IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 6 October 2023, accepted 13 November 2023, date of publication 16 November 2023,
date of current version 29 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333866

SURVEY

Audio Deepfake Approaches

OUSAMA A. SHAABAN[1], REMZI YILDIRIM[2], AND ABUBAKER A. ALGUTTAR[1]
[1]Graduate School of Natural and Applied Sciences, Ankara Yıldırım Beyazıt University, 06760 Ankara, Turkey
[2]Department of Computer Engineering, Ankara Yıldırım Beyazıt University, 06760 Ankara, Turkey

Corresponding author: Ousama A. Shaaban (205101405@aybu.edu.tr)

ABSTRACT This paper presents a review of techniques involved in the creation and detection of audio deepfakes, the first Section provides information about general deep fakes. In the second section, the main methods for audio deepfakes are outlined and subsequently compared. The results discuss various methods for detecting audio deepfakes, including analyzing statistical properties, examining media consistency, and utilizing machine learning and deep learning algorithms. Major methods used to detect fake audio in these studies included Support Vector Machines (SVMs), Decision Trees (DTs), Convolutional Neural Networks (CNNs), Siamese CNNs, Deep Neural Networks (DNNs), and a combination of CNNs and Recurrent Neural Networks (RNNs). The accuracy of these methods varied, with the highest accuracy being 99% for SVM and the lowest being 73.33% for DT. The Equal Error Rate (EER) was reported in a few of the studies, with the lowest being 2% for Deep-Sonar and the highest being 12.24 for DNN-HLLs. The t-DCF was also reported in some of the studies, with the Siamese CNN performing the best with a 55% improvement in min-t-DCF and EER compared to other methods.

INDEX TERMS Deepfakes, artificial intelligence, deep learning, audio deepfakes, forensics, datasets, survey.

I. INTRODUCTION

Deepfake refers to synthetic information or materials that have been developed or altered using artificial intelligence (AI) technologies, and are intended to be considered authentic. These may include audio, video, picture, and text synthesis [1].

In alternative narrowly defined deepfakes (coming out of Deep Learning (DL) and ''fake''), artificial neural network (ANN) innovations are important for manipulating media files. Software using (AI) such as FaceApp and FakeApp were used to superimpose the faces of a victim onto a video of the person's origin App, which was used to superimpose the faces of a victim onto a video of the person's origin. Create a video in which the intended recipient says or does something that the original provider does. Due to this trading system, anybody may buy or sell a newly generated appearance, chronological age, or even a new hairdo. Many concerns have been raised regarding the dissemination of these hoaxes [2].

Although deepfake technology may be used for beneficial objectives such as virtual reality and cinematography, its usage in criminal activities persists at high rates [3], [4], [5]. Over the last several years, thousands of fake films have gone viral online, and they are largely aimed at public figures and famous people. In 2017, a Reddit user by the moniker of deepfakes produced the first piece of deepfake material, a viral porn movie. Since the invention of the deepfake technology, dishonest applications have become commonplace. Soon after, more and more deepfake-based apps like FakeApp and FaceSwap appeared. The intelligent stripping software Deep Nude was released in June 2019 and immediately caused a frenzy. In addition to being a privacy risk, videos made with these apps are increasingly used to sway elections from the perspective of the public. The identification of false information is now at the forefront of concern for people, companies, and governments. with an increasing amount of research on deepfake devices.

Deepfake technology is not limited to its use in pornography, but is also utilized for a range of nefarious and unethical purposes. This includes the dissemination of false information, the instigation of political turmoil, and various forms of cybercrime.

More specifically, AI-synthesized systems that can produce convincing audios have recently been developed for

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

audio faking [6]. However, despite the fact that these tools were designed to benefit people, they have also been utilized to disseminate false information via audio [7], resulting in concern about "Audio Deepfakes." Recently referred to as audio manipulations, audio deepfakes are becoming more accessible through mobile devices and desktop computers [8]. This has resulted in widespread public concern regarding the adverse consequences of deep fakes in cybersecurity. Despite the advantages of this technology, audio deepfakes are more complex than simple text, email, or email links. It is possible for someone to utilize this as a logical-access audio-spoofing method [9], which opens the door to propaganda, slander, and even terrorism as a means of influencing public opinion. Detecting fakeness in vast quantities of audio recordings shared online every day is difficult [10]. However, politicians and governments are not immune to deep fake attacks [11]. For more than $ 243,000, scammers in 2019 exploited AI software to mimic a CEO over the phone [12]. Consequently, the legitimacy of all publicly available audio recordings should be verified to prevent the propagation of false information. Therefore, recent attention has been paid to this topic in the scientific community. It is becoming increasingly difficult to identify audio forgeries because of the emergence of three distinct types of deepfakes: those based on synthetic data, imitation audio, and replay data.

In addition, other detection methods are available for determining whether audio recordings contain real or fake speech. Several DL and Machine Learning (ML) models have been developed to detect fake audios using various approaches.

There are still many gaps in current algorithms [13]. Therefore, additional research is essential to enhance the detection capability of Audio Deepfakes and address the deficiencies identified in the existing literature. It has become more difficult to identify audio deepfakes owing to the emergence of new forms such as those based on synthetic imitation and replay, as discussed below.

With the advent of cutting-edge tools and DL approaches, Audio deepfake detection has become an important field of study. Currently, DL approaches have failed to compensate for these limitations. Therefore, further research is needed to determine which aspects of Audio Deepfake detection require improvement. In addition, imitated and synthetically produced audio-detection approaches have not been examined in the literature. We believe that this was a significant difference in the present study.

When evaluated on a publicly available dataset, the effectiveness of deepfake detection remained unchanged at 82.56% in recent years [14]. This is despite the fact that deepfake creation has seen significant improvement in recent years. Although this performance boost is substantial from a scholarly perspective, it is insufficient for real-world application. Recently, two major obstacles have emerged that make it crucial to consider the interpretability of deep fake detection: lower detection accuracy and increased target range. However, the current study on comprehensible deepfake detection is confined to visual deepfake detection [15], therefore it is not very broad.

Many deepfake detection strategies have emerged as a result of the increased focus on the topic of deepfake detection by academics and specialists in recent years as a means of combating these dangers. In addition, research into the existing literature on detection strategies and performance evaluation is underway. However, the scientific community and practitioners may benefit from a more in-depth study in this field that summarizes information on deepfake from all perspectives, including accessible datasets (something that has been significantly lacking in prior surveys).

This review provides a detailed analysis of audio deepfake detection techniques, along with generative approaches. The key contributions include:

a) To provide researchers with an overview of different audio methods for generating and detecting audio deepfakes.
b) Update the reader on what is new and noteworthy in the world of audio deepfakes, including techniques, tools, regulations, and problems.
c) Help the reader realize the probable effects of audio deepfakes.
d) Provide a guidance for the research community to comprehend future audio deepfake developments.

This article is structured as follows: In Section I, we present an introduction to general deepfakes, setting the foundation for the subsequent sections. Section II delves into audio deepfakes, outlining and comparing the main methods employed in their creation and detection. Moving to Section III, we explore various datasets that play a crucial role in the development and evaluation of audio deepfake detection techniques. Finally, Section IV offers a conclusion and discussion, summarizing the limitations identified and outlining potential directions for future research.

Deepfakes can be classified into four main categories: Text, Image, Video, and Audio. While most scientists are preoccupied with investigating deepfakes in videos, the other types of deepfakes must also receive a wide range of attention due to the comprehensive advances in creating these types of deepfakes, as shown in FIGURE 1:

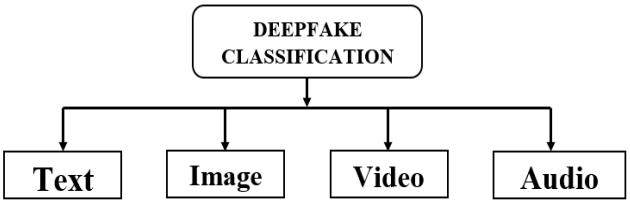


FIGURE 1. Deepfake classification [1].

Due to the rise of social media and digitalization, fake news has become a prevalent issue, challenging conventional definitions of news [2]. False information, presented as fact, is widely disseminated on online platforms [16]. Zhou and Zafarani [17] define false news as deliberately publishing incorrect materials that can be debunked by

fact-checking [5]. Previous research shows that people of all ages and backgrounds struggle to identify false news [18]. During times of uncertainty, like the COVID-19 pandemic, false rumors spread rapidly on social media, impacting public perception [19]. This phenomenon affects various aspects of life, including election campaigns, healthcare, and the economy [20], [21]. Detecting fake text is a complex task [22]. GROVER, a text generation method using GPT-2, can create highly convincing fake news [25]. Some studies employ transformer-based algorithms to identify fraudulent text on social media [24], [25]. This study investigates the detection of brief deepfake text samples from Twitter using dynamic model adjustments and a specialized BERT model [32].

Image deepfakes encompass three primary types. Firstly, there's Faceswap, widely popularized by Snapchat, allowing users to modify facial features in photographs for playful transformations [1]. Secondly, Synthesis techniques, powered by generative adversarial networks (GANs), have revolutionized image creation, with models like NVIDIA 112 generating countless variations of images [26]. Lastly, Editing, including AI-driven methods, enables significant image alterations [26]. Detecting fake images has been a focus of research, employing algorithms like k-NN, LDA, and SVM [27]. Additionally, Face-Aware Liquify in Adobe Photoshop and human artist modifications have been used [28]. Detection methods have evolved, employing supervised and unsupervised scenarios, and datasets such as StyleGAN-generated faces and iFakeFaceDB have been employed [29]. Innovations like ''facial X-ray'' [30] and attention-based CNN models [31] enhance detection capabilities, achieving impressive accuracy rates.

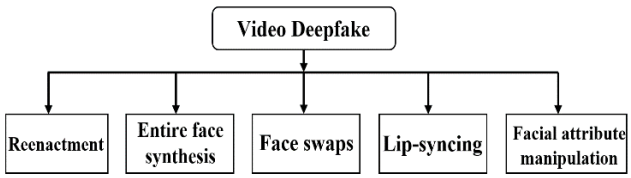


FIGURE 2. Video Deepfake classification.

Video deepfakes encompass five main categories based on the degree of manipulation as shown in FIGURE 2. These categories include face swaps, face reenactment, lip-syncing, full-face synthesis, and facial attribute manipulation. Face reenactment, for instance, manipulates facial expressions by emulating the movements of a reference actor, often used for post-production modifications in films and video games [32]. Various techniques, such as 3D facial modeling and real-time RGB-D sensor-based methods, have been employed to enhance the realism of these manipulations [33].

Lip-syncing synchronizes mouth movements with audio stimuli, essential for effective communication and accessibility for individuals with hearing impairments [11]. Techniques like Recurrent Neural Networks (RNNs) have been utilized to achieve this synchronization, while GAN-based models have improved accuracy [34].

Facial attribute manipulation alters facial features like identity or expression in images or videos, with methods like StarGAN-v2 and AttGAN producing convincing results [35], [36], [37].

Detecting video deepfakes has involved diverse approaches, from analyzing frame boundaries to utilizing CNNs and attention mechanisms, each with its strengths and limitations [38], [39], [40], [76], [77], [79], [80]. Researchers have also employed capsule networks [45] and EfficientNetB4 [46] for identification purposes. Bondi et al. examined the performance of EfficientNetB4 using multiple datasets and found that triplet loss delivered exceptional results [47].

II. AUDIO DEEPFAKES

The technology of deepfakes has been implemented in the realm of audio, specifically in the context of audio assistants and other computer-generated audios that are becoming more ubiquitous in our daily routines [48]. The utilization of artificially generated or modified audio information poses a significant threat to society, as it has the potential to generate issues of trust when individuals are incapable of distinguishing between authentic and counterfeit material [49].

There are three different varieties of speech: voiced, unvoiced, and silent. Voiced speech contains a limited amount of energy and a periodic sequence of impulses, whereas unvoiced speech consists of random, no periodic noise-like patterns. Silence, on the other hand, refers to the duration during which there are no significant signals.

Numerous linguistic factors, including formants, are used to analyses and categories utterances. Formants are frequencies where energy is densely packed, resulting in a spectral peak. In typical human speech, formants range from three to five and are categorized according to their increasing frequency. The first three formants are crucial to both voiced and unvoiced speech and are frequently employed as surrogates.

The term ''anti-forensics'' has been recently incorporated into the language of digital forensics. Although there is no consensus on the precise meaning of the term, Rogers has proposed a definition of anti-forensics as encompassing activities that aim to undermine the presence, amount, or authenticity of evidence at a crime scene, or to obstruct the examination and interpretation of such evidence during an investigation [50].

Another definition of anti-forensics is ''an attempt to prevent the recognition, collection, collation, and validation of digital data'' and established four categories of anti-forensics: data concealment, data deletion, data generation prevention, and new approaches. For example, transformation methods may be performed by malicious or rootkit-shared libraries that abuse system calls or change data during the construction process by using runtime links [50]. Practically speaking, it is the ''application of the scientific approach to digital media in order to invalidate factual material for court examination.'' [51].

"Anti-forensics" is, in general, a "collection of procedures and actions performed by a person with the intent to impede the digital investigative process" [51].

Over the past ten years, digital multimedia forensics have garnered significant interest. Most studies have concentrated on the detection of image forgery [52], with document forgery detection accounting for a large proportion [53]. However, detection of digital audio forgeries has received little attention. Before using any data as evidence in multimedia forensics, it is critical to verify both the originality and integrity of the material already in possession. The basic objective of audio forensics is to authenticate the audio by determining whether it was fake, and to identify the person or persons who were really speaking.

There are a variety of possible purposes, such as presenting it as proof in a legal proceeding or putting an end to rumors that have spread through social media or paparazzi. Digital impersonation refers to the production of speech in such a manner as to mislead humans or computers into believing that speech originates from a reliable and genuine source, thereby causing loss to society or the economy.

Synthesizing speech and altering the speaker's tone are both possible using audio-specific deep-learning techniques [54], [55]. Audio waveforms, spectrograms (which integrate information from the frequency and time domains), and other acoustic features are commonly analyzed in audio forensics to identify artificially produced or modified audio clips. The amplitude of the time-varying audio stream was analyzed using waveform-based techniques.

In [56], the authors suggested a time-domain Artificial audio Detection Network with numerous blocks, similar to those seen in ResNet and Inception networks. In [57] the study introduced a technique for detecting fabricated speech, which employs a convolutional recurrent neural network (CRNN). Rather than relying on image-based methodologies, this technique directly converts audio signal spectrograms and utilizes computer vision techniques for analysis. The spectrogram illustrates the frequency and intensity of the audio source over time.

A Mel spectrogram, which represents frequencies in megahertz, is a variation of the spectrogram [58]. For the detection of synthetic speech, Bartusiak et al. utilized a (CNN) and convolution transformer [59], [60] in combination with normalized grayscale spectrograms of the audio stream. The authors of [61] used melspectrograms to train a spatial transform network and a temporal CNN. Audio characteristics are coefficients and other values derived from the transformation process [62]. Two examples of such features are cepstral coefficients at constant Q and Mel frequencies [63], [64].

A copy-move attack can be detected by dividing an audio stream into segments, and comparing the audio characteristics of each segment, such as delta-MFCC [63], mel-frequency cepstral coefficients (MFCC) [63], and pitch [65]. using Pearson correlation coefficient [65] Higher degrees of resemblance indicate a copy-and-paste attack. Hassan and Javed [66] utilized a (RNN) to evaluate (MFCC), Gammatone Cepstral Coefficient (GTCC), spectral flux, and spectral centroid as potential markers of artificial noise in their study. Das et al. and Li et al. [67], [68]. have suggested the utilization of Inverted Constant-Q Coefficient (ICQC), Inverted Constant-Q Cepstral Coefficient (ICQCC), and Long-term Variable Q transform (L-VQT) techniques to identify synthetic music. Additionally, the authors of [106] explored the use of a Res2Net network trained on log-power magnitude spectrograms, Linear Frequency Cepstral Coefficients (LFCC), and Constant-Q Transform (CQT) for identifying synthetic audio.

A. AUDIO DEEPFAKE GENERATION METHODS

One type of deepfake is AI-generated audio manipulation, which can clone a human audio and portray it as having said something controversial that it never really utters. Fake audios participants [70], [71]. Synthetic audios are suitable for several applications such as automatic audio labelling for combine AI with human editing. For instance, speech that are similar to genuine speech have become a reality because of the recent breakthroughs in AI generation methods for audio television plus films, AI assistance, and individualized synthetic audios for persons with vocal issues. Additionally, fake/synthetic audios have become a growing challenge to vocal biometrics [72] It could be used for evil purposes such as spreading propaganda, spreading false news, or even committing fraud.

The synthesis of higher quality audios may synthesis and cloning. These algorithms may produce highly convincing and identically voicing synthetic audio in response to the text or utterances of target synthesis models powered by neural networks such as Google's Wavenet [73] and Tacotron [74] or AdobeVoco [75] may generate convincing counterfeit audios that sound like the target's vocal audio; for example, the software for editing audio [76] might be used to produce more powerful audios by combining natural and synthetic audio sources.

In addition to images, recent developments in AI-generated synthetic audios have enabled the production of incredibly convincing false films [11]. Such advances in audio synthesis have already demonstrated their capacity to produce convincing and natural-voicing acoustic deepfakes, thereby presenting significant threats to civilization [77]. Deepfake movie appeal and destructive impact can be increased by incorporating fake audio and visual manipulation [11]. These synthetic discourses lack features of audio quality linked to the identification of the target, which include expressiveness, roughness, breathing, tension, and emotion [78].

AI researchers are attempting to solve these issues to enable machines to mimic human speech in terms of how it sounds and how easily it can be understood. (TTS) Synthesis and (VC) are the only methods used to produce audio deepfakes. A TTS (Text-to-speech) synthesizer is a piece of software that can mimic the speech of any speaker [79]. a methodology utilized to transform an audio waveform

TABLE 1. List of tools, applications, and open-source projects which synthesize audiovisual deepfakes.

Tool	Technique	Type	Reference
Respeecher	Combining conventional electronic signal processing technologies with unique deep generative modeling methods	Commercial Application	https://www.respeecher.com/
Overdub	AI based	Commercial	https://www.descript.com/overdub
ResembleAI	AI based	Commercial	https://www.resemble.ai/
Voicery	DL with proprietary AI	Commercial	https://www.voicery.com/
VoiceApp	Proprietary (AI-based)	Mobile app	https://www.zoezi.com/
SV2TTS	Multi-stage LSTM with a standardized loss function	Free and open source	https://github.com/CorentinJ/real-time-voice-Cloning

originating from a source audio into one that emulates the vocal characteristics of a selected speaker [80].

The VC system uses an audio clip collected from the user as its source, and generates a radically false audio file for the target subject. It maintains the grammatical and phonetic aspects of the original sentence, while emphasizing its quality and likeness to the target speaker. Both VC synthesis and TTS pose significant dangers, because they create audios that are nearly unidentifiable in human speech. In addition, duplicated replays initiate assaults. Vocal biometric devices are of concern because improved audio synthesis algorithms can create audios similar to those received by loudspeakers [81].

This section summarizes the recent advances in speech synthesis, including TTS, audio conversion, and detection approaches Table 1 shows a list of tools, applications, and open-source projects which synthesize audiovisual deepfakes.

1) TTS (TEXT-TO-SPEECH) AUDIO SYNTHESIS

TTS, which has been around for over a decade, is a system that uses text input to generate an artificial audio, and allows the use of an audio for better interaction between humans and computers. The first investigations on TTS synthesis were conducted using audio concatenation and parameter estimation. The concatenated text-to-speech (TTS) technique entails the fragmentation of superior audio recordings into smaller units, which are subsequently reassembled to generate a novel speech pattern. Nonetheless, owing to the absence of advancement and lucidity in this methodology throughout the years, its appeal has diminished. Parametric models differ from other models in that they employ a mapping technique to convert text into basic speech features, which are subsequently transformed into an audio stream using vocoders. Subsequently, DL became an important audio synthesis approach, resulting in a much higher degree of audio quality. These include neural audio encoders [82], [83], auto encoder [84], autoregressive models [74], [85], [86], GAN [87], [88] as well as other emerging technologies. References [89] and [90] have contributed to the fast expansion of the speech synthesis industry. Figure 3 shows the rationale underlying recent Text-to-speech (TTS) techniques.

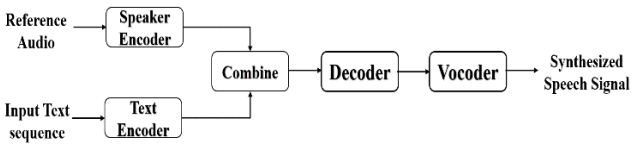


FIGURE 3. Workflow diagram of the most recent TTS systems.

WaveNet is a primary advancement in audio and speech synthesis. [85], Tacotron [74], and DeepVoice3 [91] can produce realistic synthetic audios from text inputs to enhance the interaction between people and robots. WaveNet, Developed in [85], which is the result of the creation of a pixelCNN, was developed in [121]. conditional image [92]. WaveNet models employ acoustic information such as spectrograms to convert raw audio waves throughout a generating frame based on real audio data. Parallel WaveNet technology was developed to enhance the sampling efficiency and provide high-quality audio streams [93]. An additional DL that depends on a WaveNet variation called Deep Voice1 [94] is available by swapping an associated NN template for any module with an audio source, speech synthesizer, or text processing interface. There is no actual end-to-end audio-recognition technology; however, each module is individually taught. Google coined the word ‘‘taciturn’’ in 2017. The All-inclusive Audio Synthesis Model Taciturn can synthesize audio using textual and audio pairings, making it sufficiently versatile for use with many different types of dataset. Tacotrons, such as WaveNet, are generative systems composed of a sequence-to-sequence model, attention-based decoding, and post-processing net. However, the superior performance of the tacotron model may be accompanied by certain drawbacks. This process is repeated several times.

Model creation requires high-performance systems because of the inefficiency in including these components.

Deep audios using a combination of Tacotron and WaveNet [95] synthesize speech. Tacotron uses WaveNet to translate source text into speech by transforming it into a linear spectrum. Tacotron2 was created in [96] is a speech synthesis system that utilizes a neural network architecture consisting of an encoder network, a decoder network, and an attention mechanism to generate speech waveforms of superior quality. The encoder network is responsible for receiving textual input and producing a series of embedding that encapsulate the semantic content of the input. Subsequent to the production of embedding by the encoder, the decoder network proceeds to generate a sequence of Mel-spectrogram frames. The attention mechanism facilitates the decoder's ability to concentrate on pertinent segments of the encoder output during the production of Mel-spectrogram frames.

The Tacotron2 model has been extensively employed in various domains, including but not limited to audiobook narration, virtual assistants, and chatbots. The system has exhibited exceptional proficiency by producing speech that closely resembles human-like quality and intonation. Furthermore, Tacotron2 exhibits a high degree of adaptability and can undergo training on diverse linguistic and vocal modalities.

Notwithstanding its accomplishments, Tacotron2 encounters certain obstacles, including generating speech that sounds authentic in the presence of ambient noise and managing texts of extended length. Nevertheless, current research endeavors to tackle these concerns and enhance the efficacy of vocal synthesis systems such as Tacotron2. The researchers created DeepVoice3, an entirely convolutive spectrogram character model, to overcome the temporal complexity associated with recurrent unit-based audio synthesis models. Reference [91] shows that the Deep Voice 3 model outperforms its rivals in terms of speed because all calculations are performed in parallel. There are three main components to Deep Voice 3: The vocoder has three parts: (1) Encoder that transforms input text into a learnt internal code form; (2) a decoder that interprets autoregressively learned representations; and (3) a wholly convolutive post-processing network that predicts the parameters of the vocoder. VoiceLoop is an alternative audio synthesis model. Using a memory frame, speech was generated from audios that were not audible during the training. VoiceLoop constructs phonological storage by using an offset buffer as a matrix. Phonemes in a string of texts were converted into tiny vectors for representation. The generated phonemes were evaluated and their codes were added to form a new contextual vector.

There has been a significant amount of research put into and development of end-to-end audio synthesis models. In [91], researchers discussed various methods that can be used to construct such models. Additionally, commercial products such as Amazon AWS Polly, Baidu TTS, and Google Cloud TTS have been introduced by [131]. with the goal of achieving a high degree of similarity between the synthesized and natural audios. These products aim to achieve this similarity by using a variety of techniques. These systems are becoming more and more popular, and they are currently utilized for a wide variety of applications including chatbots, virtual assistants, and audiobooks.

Text-to-speech (TTS) systems of the modern era are capable of effectively converting written text into speech that sounds natural and possesses particular characteristics. Researchers have been able to develop speech models that can replicate the audio of a particular speaker with remarkable accuracy, even with only a small number of reference samples to use as a guide. This has been made possible by the development of neural network models. This has ushered in a new era of real-time audio cloning technology; in which it is now possible to synthesize the audio of a person in real-time using only a few seconds of their speech as input.

This has opened the door to a wide range of applications for the technology. This type of technology has a wide variety of applications in the real world, ranging from individualized audio assistants to assistive technology for people who have difficulties communicating through speech [89], [97]. Audio synthesis systems do not seek to imitate a person's distinctive speech features, whereas speech cloning systems do [98]. ISpeech3, VoiceApp2, and Overdub1 are only a few examples of AI-powered Audio-cloning platforms that make this technology publicly available by generating synthetic false audios that mimic targeted speech.

The authors of [89] Developed a TTS system that relies on Tacotron 2, which can synthesize the Audios of several speakers, even those not present throughout training. Three neural networks, each trained separately, constitute the framework. Synthetic speech correctly imitates a target speaker's Audio but not its prosody.

In [107], the authors recommended two Deep Voice 3 modules: speaker encoding and speaker adaptation. Speaker adaptability was prioritized in the framework to produce audio for several channels. To encode speakers, a second model was trained to use the multi speaker generative framework to determine new speaker embedding.

In [133], researchers unveiled a speech cloning algorithm that, given a text input or audio waveform of a speaker as input, can synthesize an audio that sounds similar to the one that the system is supposed to mimic. The architecture incorporates a neural vocoder, in addition to text and audio encoders and decoders. The speech-generation model is instructed by a representation disentangled from the speaker and the approach is jointly trained using latent linguistic characteristics. Cloning a speaker's audio takes approximately five minutes, but the final product is of exceptional quality and is similar to the original speaker.

The authors of [99] have suggested a meta-learning approach to enhance the efficacy of audio commands. This approach involves the integration of a WaveNet model that can operate with restricted data. The initial stage of this

TABLE 2. Overview of the latest audio synthesis methods.

Methods	Technique	Features	Used Dataset	Weak points
DeepVoice 2 [95]	RNN	Deep features	Private (24.6 hrs.)	▪ Training the model might be expensive.
Taciturn[74]	Encoder-Decoder with RNN	Deep features	Private (20 hrs.) VCTK (44 hours) LibriSpeech ASR (820 hrs.)	▪ Does not generalized well for unseen samples.
DeepVoice3 [91]	Encoder-decoder	Deep features	VCTK (44 hours)	▪ Training the model might be expensive.
DeepVoice 1[94]	Deep neural networks	linguistic attributes	Private (20 hrs.)	▪ Individual module training results in a cumulative inaccuracy in synthesised speech.
Parallel WaveNet [93]	Feed-forward neural network with dilated causal convolutions	linguistic attributes	Personal	▪ A substantial quantity of data specialising in the target's Speech is needed for training.
Tacotron2[96]	Encoder-decoder	linguistic attributes	ATR Ximera Japanese corpus (46.9 hrs.)	▪ Problems with time-delayed speech synthesis
Arik et al. [70]	Encoder-decoder	Mel spectrograms	LibSpeech (820 hrs.) VCTK (44 hours)	▪ Poor results when trying to generate speech for several speakers from low-quality audio.
Jia et al. [89]	Encoder-decoder	Mel spectrograms	LibSpeech (436 hrs.) VCTK (44 hours)	▪ Naturalness below human level is not achieved. ▪ Synthesized speech lacks the target accent and prosody.
Luong et al. [98]	Encoder-decoder	Mel spectrograms	LibSpeech (245 hrs.) VCTK(44 hours)	▪ Poor efficiency while dealing with loud audio samples
Cong et al. [90]	Encoder-decoder	Mel spectrograms	MULTI-SPK CHiME-4	▪ Inability to effectively synthesize the speech of a desired speaker
Chen et al. [99]	Encoder + deep neural network	Mel spectrograms	LibSpeech (820 hrs.) private	▪ Poor results when using an inferior audio source
WaveNet[85]	DNN	▪ Linguist ▪ Baseline Frequency (log F0)	VCTK (44 hrs.)	▪ Complex and time-consuming to compute
VoiceLoop[86]	Integral neural network	Advanced acoustic capabilities in 63 dimensions	VCTK (44 hrs.) Private	▪ Lacking eco- relevance

methodology entails the computation of speaker adaptation through the refinement of audio embedding. Subsequently, the embedding vectors of novel speakers can be forecasted utilizing a parametric methodology that is not dependent on textual data. This approach may prove advantageous in situations where there is a scarcity of data and prompt adjustment to novel speakers is imperative.

The findings of this investigation exhibit the efficacy of the suggested methodology in producing superior synthetic vocalizations for diverse speakers. An additional encoding network must be constructed to achieve this. This technique works well when training high-quality, clean data. The quality of the synthesized speech is reduced when background noise is present during the encoding process. In [125], The researchers presented a multi-speaker sequence-to-sequence model. The model utilizes domain-specific training data to reconstruct the speech of a target speaker from a restricted number of noisy input samples. The methodology entails the process of instructing the model using a dataset that encompasses audio samples from various speakers. Subsequently, the model is fine-tuned on a smaller dataset that comprises samples from the specific target speaker. The model's output exhibits the ability to produce synthetic speech of superior quality that bears a striking resemblance to the natural audio of the target speaker, despite the presence of restricted training data. The aforementioned methodology exhibits the capability to facilitate a diverse array of implementations, such as audio replication and audio transformation, while necessitating minimal data prerequisites. The results showed that the artificial speech became more lifelike. Consequently, creating convincingly equivalent synthetic speech from a limited amount of poor-quality audio data remains a challenge. Table 2 summarizes the sophisticated audio synthesis approaches.

2) VOICE CONVERSION (VC)

Voice synthesis using VC makes the source voice appear more like the desired voice while keeping the original grammar intact. VC is used for various purposes in the entertainment sector, including expressive audio synthesis, individualized speech assistance for individuals with hearing impairments, and audio dubbing [80]. Recent advances in anti-spoofing technology for automatic speaker recognition [72] have included VC systems for generating fake data [72], [100]. Audio control relies on more advanced aspects of speech such as timbre and prosody. Prosody is concerned with suprasegmentally features, such as pitch, amplitude, stress, and duration, whereas audio timber focuses on the spectral characteristics of the auditory system through phonation. Several voice conversion competitions (VCCs) have been organized to promote research on voice conversion methods and improve the accuracy of existing methods, [72], [100] Scholars in the domain of speech conversion have been investigating techniques to enhance the caliber of speech conversion by utilizing both parallel and non-parallel data. The Voice Conversion Challenge (VCC) is a technique that aims to convert input audio to output speech through the integration of parallel and non-parallel training data, as described in [72] and [101]. The objective of the VCC was to mitigate the constraints associated with conventional parallel training data by investigating the potential of non-parallel data, which is frequently more prevalent in practical settings. In [136], significant endeavors were undertaken to devise techniques for cross-lingual voice conversion (VC), which pertains to the process of transforming recorded speech from one language to another.

The research was centered on non-parallel training data and encompassed a diverse array of languages in order to tackle the difficulty of cross-lingual voice conversion. The findings indicate encouraging enhancements in the standard of speech conversion, highlighting the viability of integrating non-parallel data with conventional parallel training techniques. Previous research has shown that VC methods depend on spectrum mapping with paired training data, and require the use of audio samples from both target and source audios that share common linguistic content. Gaussian Mixture Model GMM-based techniques [26], [102], regression using partial least squares [103], exemplar-based techniques [104], and parallel spectral modeling [105], [106] have been suggested. These [102], [104] are ''shallow'' VC approaches that could directly modify the spectral features of the source audio in its native feature space [105] To capture temporal correlation in an audio stream. Researchers previously proposed an RNN-based speaker-dependent sequential approach.

In [106] and [143], the deep bidirectional LSTM (DBLSTM) methodology made it feasible to extract long-range contextual data while producing high-quality transformed audios using DNN-based techniques. This is made possible by the fact that DBLSTM is a technique. In [105] and [106], feature representations were efficiently learned for simultaneous VC feature mapping. Parallel training requires a large number of source and target spoken phrase samples, which is impractical for practical application. Researchers have proposed VC algorithms for nonparallel (unpaired) training data as a means to achieve voice conversion for speakers of diverse languages. The algorithms in question endeavor to enhance the quality of speech conversion through the utilization of both parallel and non-parallel data.

In a particular research endeavor [136], a significant endeavor was undertaken to devise techniques for cross-lingual voice conversion (VC) utilizing non-parallel training data and a diverse set of languages. This process entails the translation of one language into another through the use of recorded speech. Robust value creation approaches, such as neural network-based [108], vocoder [109], [110], GAN [111], [112], and VAE [113], [114] have been developed to aid in the modeling of non-parallel spectral data.

Techniques based on auto encoders attempt to learn how to modify speaker identification independently of the linguistic content. In [114], the quality of learned representations was compared using various auto encoding techniques. Whenever WaveNet and Vector Quantized VAE are used together, it was found that, [85] The decoder enhances the preservation of speaker-invariant language content and recovers rejected information. Owing to the dimensionality reduction bottleneck, VAE/GAN-based techniques over smooth the transformed features, resulting in voice conversion with audio buzz.,

Recent GAN-based approaches such as VAW-GAN [115], CycleGAN [111], [116], and StarGAN [153] aim to produce high-quality converted speech. Studies [117], [118] have demonstrated superior performance in terms of sounding natural and similar to the target audience compared to other multilingual VC. Therefore, performance is reliant on the presence of a speaker, and diminishes for unseen speakers. Owing to their capacity to create human-like speech, neural vocoders have surpassed other vocoding technologies and become the standard for audio synthesis in recent years [91]. The vocoder shows the ability to acquire and produce audio waves that bear a striking resemblance to the distinct acoustic characteristics of the speaker.

Research [110] examined the performance of a variety of vocoders and determined that parallel-WaveGAN performed the best. Using acoustic properties, [119] effectively simulated the transmission of human speech data over an IP (VC). Nevertheless, there is scope for improvement in addressing unidentified louder speakers [71] Using AttS2S-VC, [120] Cotatron, [121] and VTN, [122] researchers can directly synthesize speech from text labels using three modern VC techniques based on TTS by detecting aligned linguistic features from the source speech. By doing so, we know that neither the source nor destination speaker's identity will change throughout the conversion process. Unfortunately,

TABLE 3. Overview of the various methods for detecting deepfake audio.

Authors	Technique	Features	Used Dataset	Weak points
Zhang et al. [131]	ResNet- 18+OC-softmax	Deep features	ASVspoof 2019	▪ Performance declines on VC.
Gomez-Alanis et al. [132]	LCG- RNN	Deep features	ASVspoof 2019	▪ Unseen attacks aren't generalized
Hua et al. [56]	Res-TSSDNet	Deep features	ASVspoof 2019	▪ Complex in terms of computation
Jiang et al. [133]	CNN	Deep features	ASVspoof 2019	▪ performance not adequate
Li et al. [69]	Res2Net	CQT	ASVspoof 2019	▪ Weakness in generalization; needs work
Das et al. [134]	LCNN	CQT	ASVspoof 2019	▪ Training data needed
Aljasem et al. [135]	Asymmetric bagging	MFCC, GTCC, ALTP, and spectral	ASVspoof 2019	▪ performance not adequate
Ma et al. [136]	CNN	60-D LFCC	ASVspoof 2019	▪ Performance declines on noisy samples
Gao et al.[137]	ResNet	2D-DCT	ASVspoof 2019	▪ Performance declines on noisy samples
Aravind et al. [138]	ResNet34	Mel-spectrogram	ASVspoof 2019	▪ performance not adequate
Chen et al. [9]	ResNet	60-dimensional LFB	ASVspoof 2019	▪ costly strategy
Huang et al. [139]	DenseNet- BiLSTM	LFBank	ASVspoof 2019	▪ costly strategy.
Wu et al. [140]	LCNN	Genuine speech	ASVspoof 2019	▪ Failure to identify replay attacks recognition.
Zhang et al. [141]	TEResNet	Spectrum	ASVspoof2019 Fake-or-Real [142]	▪ Training data needed
Wang et al. [143]	DNN	Deep features	Fake-or-Real [142]	▪ Evaluation required on complex dataset
Monteiro et al. [144]	LCNN/ResNet	Spectral	Propriety	▪ Results must be assessed using a standard dataset.
Singh et al. [145]	Quadratic SVM	Bispectral and mel-cepstral	Propriety	▪ Needs assessment on a massive scale dataset
AlBadawy et al. [146]	logistic regression classifier	Bispectral	Propriety	▪ Performance may deteriorate with high-quality speech samples
Yi et al. [147]	GMM/LCNN	CQCC	Propriety	▪ The performance of partially generated audio clips diminishes

these strategies rely on text labels, which are not always easily accessible.

There have been recent attempts at ''one-and-done'' VC techniques [123], [160]. Unlike prior methodologies, the process of training few-shot voice conversion models does not necessitate direct access to the source and target speaker data samples. Merely one statement from each speaker suffices for the conversion procedure. The speech of the source speaker is utilized to derive a speaker embedding, which is subsequently employed to produce the converted speech. Notwithstanding recent progress, the few-shot voice conversion techniques for speakers who have not been previously encountered still encounter obstacles in attaining dependable performance. [125]. This is largely because speaker embedding generated from a single unseen speaker's speech is insufficient [126]. This has a noticeable effect on the dependability of the one-shot conversions. The additional effort [127], [128] of speaker identities is concealed during training using zero-shot VC and the model does not need to be retrained.

The speaker encoder breaks down data about the speaker's delivery into individual ''embeddings'' for style and substance, while the decoder uses these ''embeddings'' to construct audio clips. The zero-shot VC scenario is interesting because it does not require collection or adjustment of parameters or data for adaptation. However, adaptation falls short, especially in situations in which both the goal and source speakers are invisible, vastly dissimilar, and very loud [125].

B. AUDIO DEEPFAKE DETECTION METHODS

Due to recent advancements in TTS [126], [129] and VC [125] techniques, deepfakes in audio pose a growing threat to audio biometric interfaces and society. Previous research has not fully addressed the detection of synthetic speech [130], but DL methods, such as CNNs, RNNs, and LSTMs, show promise in detecting deep fakes by analyzing spectral content, pitch, and time-frequency patterns. The use of these methods holds great potential for preventing the spread of audio deep fakes. This section examines the methodologies for detecting audio deepfakes.

In the previous TABLE 3, an overview of various methods for detecting deepfake audio was provided. Two primary

categories can be established for the techniques.: handcrafted techniques and DL techniques. Handcrafted techniques involve manually designing and implementing algorithms to detect deepfake audio, while DL techniques utilize neural networks to automatically learn patterns in the audio data and detect deepfakes. In the following text, we will delve deeper into each of these categories and discuss the specific methods used

1) HANDCRAFTED TECHNIQUES

Yi et al. [131] proposed a technique for identifying audio content that has been modified using TTS synthetic speech

recognition, which may be trained using GMM and LCNN classifiers using constant Q-cepstral coefficients (CQCC), which require handcrafted features. Although this technique performed better with completely synthesized audio, its performance progressively declined with the partially generated audio samples [106]. Res2Net is a modified version of the ResNet. They assessed the model using a variety of acoustic properties and determined that CQT features provided the best results. This model performs better at detecting audio tampering; Nonetheless, there is room for further enhancement of its capacity for generalization.

In [132], utilized a combination of mel-spectrogram features and ResNet-34 for the purpose of detecting counterfeit speech. Despite the success of this approach, there is room for further improvement. The authors Monteiro et al. [133] have presented their research findings. An approach utilizing ensembles was employed to distinguish between authentic and synthetic speech. Deep learning models, specifically LCNNs and ResNets, were utilized to compute deep attributes, which were subsequently combined to achieve this objective. Despite the robustness of the false speech detection, it is crucial to evaluate this model on a representative dataset.

A method for identifying counterfeit speeches was devised by Gao et al. [134] which relies on the detection of such inconsistencies. A residual network was trained to identify altered speech through the utilization of a global 2D-DCT feature. Although the model exhibited a higher degree of generalization, its performance deteriorated when noisy data was used. An artificial speech detection model based on the ResNet network and transformer encoder was developed by Zhang et al. [135] (TEResNet). The initial stage involved the utilization of a transformer encoder to build context-specific renderings of an acoustic key point by analyzing the correlation between the frames of the audio input. Subsequently, a residual network was trained using the determined key points to differentiate between unaltered and changed speeches. This study demonstrates improved effectiveness in detecting bogus audio but requires substantial training data.

In the study [136] conducted by Das et al. they developed a technique for determining whether an individual's speech has been altered. First, a signal commanding approach was utilized to boost the variety of the training data. Subsequently, the data that was gathered was employed to produce CQT characteristics, which were subsequently utilized for training the LCNN classifier. Although this approach improves the accuracy of detecting counterfeit audio, it requires a substantial quantity of training data.

The detection of copied conversations was proposed by Aljasem et al. [137] through the utilization of a technique that relies on handcrafted features. At the outset, sign-modified acoustic local ternary patterns were utilized to extract features from the input data. The knowledge that was acquired was subsequently utilized to develop classifiers based on asymmetrical bagging technique for the purpose of discriminating between authentic and artificially generated speeches. The aforementioned technique exhibits resilience towards high-volume cloned vocal playback assaults. Nevertheless, it necessitates additional refinement with regards to its efficiency.

Ma et al. [138] introduced a method based on continuous learning to improve the ability of modified speech detection systems to generalize. The learning capabilities of the model were enhanced by adding a loss function to distill accumulated information. Although this technique is computationally efficient and capable of detecting previously undiscovered spoofing operations, its performance with noisy data has not been examined.

Borrelli et al. [139] included both short- and long-term bicoherent characteristics in their study. Three classifiers were trained using the gathered features: linear (SVM), radial basis function (RBF), and random forest (SVM). This technique achieves the highest degree of precision when an SVM classifier is used. However, because this is a manual process, it cannot be used to hide procedures. When analyzing GAN-generated audio samples, [140] researchers have employed bispectral analysis to identify the unique spectral correlations.

Similarly, in [141], utilized bispectral and mel-cepstral analyses to identify the missing robust power elements in counterfeit speech. The aforementioned characteristics were employed for the purpose of instructing diverse classifiers grounded on ML, among which a Quadratic Support Vector Machine (SVM) exhibited the most superior performance. These strategies [140], [141] are unaffected by TTS synthetic audio but may miss the audio synthesis of superior-quality Malik and Changalvala [142] suggested using a CNN to identify clone speech.

First, audio samples were transformed into spectrograms, which were then used to calculate the deep features and categorize the actual and false speech samples using CNN architecture. Although this method is more effective in identifying phony audio, it suffers from samples with high levels of background noise. Chen et al. [9] developed a technique that exploits DL to recognize fake audios. Audio samples were used to create linear filter banks (LFB) with 60 dimensions based on which a specialized ResNet model was trained. This study enhances the identification of bogus audios, albeit at considerable computational expense.

Huang and Pun [143] proposed a technique to detect audio spoofing. First, silences were identified by analyzing the rate and intensity of each speech signal's short-term zero crossing. Then, in the relatively high-frequency domain, the chosen sections were used to identify the LFBank critical spots. Finally, a superior DenseNet-BiLSTM framework was developed for audio manipulation detection. However, the computational cost of this method [143] for avoiding overfitting is high. Based on keypoint and light CNNs, Wu et al. [144] suggested a novel approach for detecting synthetic audio manipulations (LCNN).

The unique characteristics of human audios were used to train a (CNN) model. Alterations were made to make the emphasis distribution more similar to that of the normal speech. An LCNN was then used in combination with the modified keypoints to distinguish natural speech from artificial speech. That's because this method [144] can't be easily fooled by artificially altered audios. However, it cannot prevent assaults from using a replay, because it has no way of identifying them.

2) DEEP LEARNING FEATURES-BASED TECHNIQUES

Zhang et al. [145] showed that a (DL) strategy can be developed using OC Softmax and ResNet-18. The model was trained to identify the feature space that allowed for differentiation between the natural and modified audio samples. Despite its superiority in generalization against unknown assaults, this technique suffers from VC attacks owing to the waveform filtering.

AS shown in figure (4) the system adheres to a conventional architecture based on deep learning for the purpose of detecting audio spoofing. The characteristics of the speech are inputted into a neural network for the purpose of computing an embedding vector that corresponds to the inputted utterance. The neural network is trained to acquire knowledge of an embedding space that enables efficient differentiation between authentic audios and those produced through spoofing. Subsequently, the embedding is employed to evaluate the level of certainty regarding whether the utterance pertains to genuine speech or spoofing.

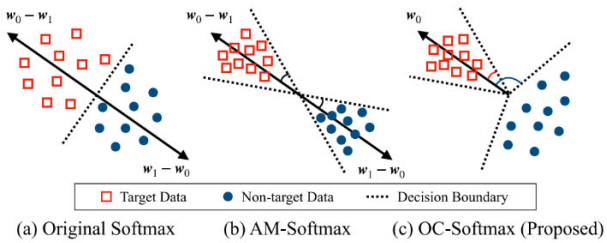


FIGURE 4. An illustration of Softmax and AM-Softmax for binary classification, alongside the proposed OC-Softmax for one-class learning. The embeddings and weight vectors presented in the illustration are not normalized [145].

The study presents a new loss function, denoted as One-class Softmax (OC-Softmax), which is designed for the purpose of detecting audio spoofing. This is juxtaposed with the frequently employed loss functions for binary classification. The OC-Softmax loss function has been developed with the purpose of condensing the authentic speech representation and segregating the instances of spoofing attacks in the embedding space.

The mathematical equations utilized by the system are as follows:

$$\mathcal{L}_S = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\boldsymbol{w}_{y_i}^T\boldsymbol{x}_i}}{e^{\boldsymbol{w}_{y_i}^T\boldsymbol{x}_i}+e^{\boldsymbol{w}_{1-y_i}^T\boldsymbol{x}_i}} = \frac{1}{N}\sum_{i=1}^{N}\log\left(1+e^{(\boldsymbol{w}_{1-y_i}-\boldsymbol{w}_{y_i})^T\boldsymbol{x}_i}\right), \quad (1)$$

The softmax loss is a loss function commonly employed in the training of classification models. The aforementioned statement pertains to the quantification of the dissimilarity between the anticipated probability distribution and the factual probability distribution. During the training process, the model aims to minimize the softmax loss, which facilitates the acquisition of the ability to accurately predict the appropriate class for every input.

$$\mathcal{L}_{AMS} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\alpha\left(\hat{\boldsymbol{w}}_{y_i}^T\hat{\boldsymbol{x}}_i-m\right)}}{e^{\alpha\left(\hat{\boldsymbol{w}}_{y_i}^T\hat{\boldsymbol{x}}_i-m\right)}+e^{\alpha\hat{\boldsymbol{w}}_{1-y_i}^T\hat{\boldsymbol{x}}_i}} = \frac{1}{N}\sum_{i=1}^{N}\log\left(1+e^{\alpha\left(m-(\hat{\boldsymbol{w}}_{y_i}-\hat{\boldsymbol{w}}_{1-y_i})^T\hat{\boldsymbol{x}}_i\right)}\right), \quad (2)$$

The AM-Softmax loss function is utilized in the training of one-class classification models. The proposed approach is a variant of the softmax loss function, which incorporates an angular margin to enhance the compactness of the embedding distributions for each class. During the training process, the AM-Softmax loss function is minimized, thereby facilitating the model's ability to differentiate between authentic and counterfeit vocalizations.

$$\mathcal{L}_{OCS} = \frac{1}{N}\sum_{i=1}^{N}\log\left(1+e^{\alpha(m_{y_i}-\hat{\boldsymbol{w}}_0\hat{\boldsymbol{x}}_i)(-1)^{y_i}}\right) \quad (3)$$

The OC-Softmax loss is a loss function utilized in the training of one-class classification models. The proposed approach involves a variation of the AM-Softmax loss function, which incorporates dual margins to enhance the compression of genuine speech and effectively isolate instances of spoofing attacks. During the training process, the model is trained to minimize the OC-Softmax loss, which in turn enhances its ability to accurately detect deepfakes.

Hua et al. [56] presented The Res-TSSDNet and Inc-TSSDNet models for the purpose of detecting synthetic speech. Both models exhibit a comparable architecture, characterized by a series of stacked ResNet-style or Inception-style blocks, fully-connected linear layers, and global max pooling. The Inc-TSSDNet utilizes dilated convolutions in its models to augment the receptive field. The training approach encompasses the preparatory phase of data, utilization of

weighted cross-entropy loss to address data imbalance, and implementation of mixup regularization to enhance generalization. The findings demonstrate the efficacy of the proposed models in relation to established techniques, as evaluated on the ASVspoof2019 dataset. An ablation study is conducted to assess the influence of the dimensions of network depth and width. According to the research, it is suggested that lighter models are able to attain a favorable balance between precision and effectiveness.

$$\mathrm{WCE}\left(\mathbf{z}, y_i\right) = -w_{y_i} \log\left(z_{y_i}\right) \tag{4}$$

The utilization of weighted cross-entropy loss is a strategy to address the issue of data imbalance, whereby the minority class is assigned a higher weight.

$$\mathrm{CE}_{\mathrm{mixup}}\left(\tilde{\mathbf{z}}, y_i, y_j\right) = \lambda \mathrm{CE}\left(\tilde{\mathbf{z}}, y_i\right) + (1-\lambda)\mathrm{CE}\left(\tilde{\mathbf{z}}, y_j\right), \tag{5}$$

The study employs the mixup regularization loss function, which integrates the cross-entropy (CE) losses of the mixed examples in the synthetic speech detection domain.

$$\tilde{x}_i = \lambda x_i + (1-\lambda)x_j, \tilde{y}_i = \lambda y_i + (1-\lambda)y_j \tag{6}$$

The regularization loss of mixup pertains to the amalgamation of training examples and labels to enhance the generalization of the model.

Wang et al. [146] devised a deep neural network (DNN) model, which they named Deep-Sonar, to identify artificially-generated counterfeit audios in speaker recognition (SR) systems. The employed methodology utilizes a stratified configuration of neural units to execute the task of classification. The Deep-Sonar system was assessed by the authors on the audios of English speakers obtained from the FoR dataset [147], The results showed a detection rate of 98.1% and an equal error rate (EER) of approximately 2%. The model's efficacy was notably impacted by the existence of noise in practical settings, leading to a reduction in precision. Wang et al. presented a noise-reduction methodology to tackle the aforementioned problem. The proposed technique yielded a 5% improvement in the model's accuracy, leading to a detection rate of 98.6% and an EER of 1.9%.

As can be seen from figure (5) The system comprises three primary constituents:

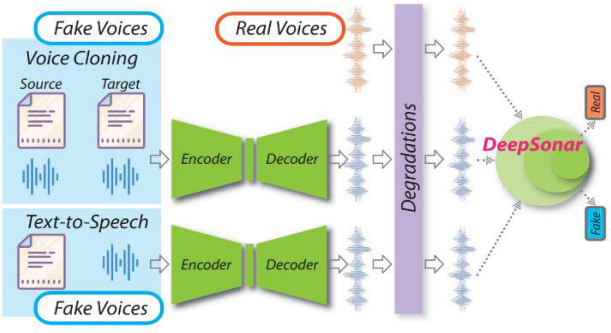


FIGURE 5. The DeepSonar system's block diagram for spotting AI-generated doppelganger voices [146].

deep neural network (DNN) was proposed to differentiate between authentic and counterfeit audios. The DNN was trained on a speaker recognition (SR) system and extracts activation patterns from the SR system for both authentic and synthetic vocalizations. A classification system was then used to distinguish between authentic and counterfeit audios based on the extracted activation patterns.

The operational process involves the initial input of authentic and counterfeit vocalizations into the SR mechanism. The SR system then generates a collection of activation patterns for every vocalization. The layer-wise neuron activation pattern extractor is then used to extract the activation patterns. Finally, the activation patterns are inputted into the classifier, which performs the task of categorizing the audios into either genuine or counterfeit.

The authors found that layer-wise neuron behaviors can be used to detect artificially-generated fake audios. The TKAN neuron coverage criterion was more effective than the ACN neuron coverage criterion because it can distinguish between real and artificial audios more effectively.

The system has reportedly shown effectiveness in identifying vocalizations that were created artificially. The system has a detection rate of 98.1% and a false alarm rate of about 2%. The system demonstrates resistance to manipulation attempts, including but not limited to audio alteration and the addition of outside noises.

the equations that utilized are:

$$\delta_l = \frac{\sum_{x \in X, i \in I} \varphi(x, i; \theta)}{|I| \cdot |X|} \tag{7}$$

computes the lth layer threshold l for the SR system.

$$ACN(l, i) = |\{x \mid \forall x \in l, \varphi(x, i; \theta) > \delta_l\}| \tag{8}$$

$$TKAN(l, i) = \{\mathrm{arg}max(\varphi(x, i; \theta), k) : x \in X \tag{9}$$

defines the neuron coverage criteria for the ACN.

defines the neuron coverage requirements for TKAN.

The findings of the authors suggest that layer-wise neuron behaviors can be used to detect artificially-generated fake audios. The TKAN neuron coverage criterion is more effective than the ACN neuron coverage criterion because it can distinguish between real and artificial audios more effectively.

In their study, Yu et al. [148] introduced a new method for scoring, referred to as Human Log-Likelihoods (HLLs), that utilizes a Deep Neural Network (DNN) classifier. In contrast to the conventional employment of (GMM) in Log-Likelihood Ratios (LLRs) scoring system, HLLs are specifically devised to augment the precision of the classification procedure. The efficacy of the HLLs approach was assessed by the authors through the utilization of the ASV Spoof Challenge 2015 dataset and the automated extraction of feature sets. The findings of the experiment indicate that the DNN-HLLs exhibited superior performance in detecting accuracy compared to GMM-LLRs, as evidenced by an Equal Error Rate (EER) of 12.24. This study provides evidence

supporting the enhanced dependability and precision of the HLLs technique in identifying falsified audio.

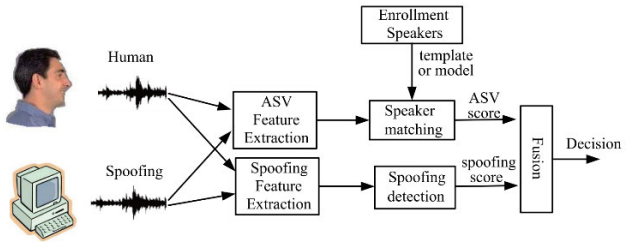


FIGURE 6. The model of spoofing detection system in an ASV system [148].

As shown in Figure (6) The system for detecting spoofing comprises three main components: feature extraction, spoofing detection, and decision-making.

The feature extraction component extracts distinctive characteristics from the audio signal input. The spoofing detection component is a deep neural network (DNN) that has been trained to differentiate between authentic and falsified audio. The decision-making component determines the authenticity of the input audio by analyzing the DNN's output.

The spoofing detection score can be computed using the log-likelihood ratio (LLR), the human-like likelihood (HLL) scoring techniques, or a combination of both. The mean (m) and standard deviation (σ) of the spoofing scores are then determined. The false rejection rate (FRR) and false acceptance rate (FAR) are also calculated.

$$\mathrm{S_{GMM}(X)} = \frac{1}{T}\sum_{i=1}^{T}\{\log P\left(X_i \mid \lambda_{\mathrm{human}}\right) - \log P\left(X_i \mid \lambda_{\mathrm{spoof}}\right)\} \tag{10}$$

The scores S1DNN(F) and S2DNN(F) are derived from a (DNN) specifically designed to discriminate against spoofing, resulting in a spoofing detection mechanism. These equations are denoted as (11) and (12). The following equations are utilized to calculate the disparity between the logarithmic posterior probabilities of authentic human speech and fraudulent spoofing techniques.

$$S1_{\mathrm{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\left\{\log\left[P\left(h \mid F_i\right)\right] - \log\left[\sum_{k=1}^{K} P\left(s_k \mid F_i\right)\right]\right\} \tag{11}$$

$$S2_{\mathrm{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\left\{\log\left[P\left(h \mid F_i\right)\right] - \log\left[max\left(P\left(s_k \mid F_i\right)\right)\right]\right\} \tag{12}$$

Equation (13) denotes that $S3_{\mathrm{DNN}}(\mathbf{F})$ is an additional metric for detecting spoofing, which is computed based on the results of the deep neural network. The approach utilizes the log-likelihood of human speech, which represents the probability of a given frame belonging to human utterance, as the metric for determining the degree of spoofing.

$$S3_{\mathrm{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\log\left(P\left(h \mid F_i\right)\right) \tag{13}$$

Equations (14) to (17) are utilized to determine the mean (m) and standard deviation (σ) of the spoofing scores (S_{HLL} and S_{LLR}) by employing the log-likelihood ratio (LLR) and human-like likelihood (HLL) scoring techniques.

$$m\_S_{\mathrm{HLL}} = E\left[y_1\right] \tag{14}$$

$$\sigma\_S_{\mathrm{HLL}} = \sqrt{\left(E\left[y_1^2\right] - E\left[y_1\right]^2\right)/T} \tag{15}$$

$$m\_S_{\mathrm{LLR}} = E\left[y_2\right] \tag{16}$$

$$\sigma\_S_{\mathrm{LLR}} = \sqrt{\left(E\left[y_2^2\right] - E\left[y_2\right]^2\right)/T} \tag{17}$$

Equations (18) and (19) denote FRR(θ) and FAR(θ) as the measures of false rejection rate and false acceptance rate, correspondingly, at a specific threshold value of θ. The cumulative distribution functions of the normal distribution are utilized in these equations to estimate FRR and FAR.

$$\mathrm{FRR}(\theta) = \mathrm{CDF}\left(\theta \mid m_h, \sigma_h\right) \tag{18}$$

$$\mathrm{FAR}(\theta) = 1 - CDF\left(\theta \mid m_s, \sigma_s\right) \tag{19}$$

Authors of [149] build a model using a Light Convolutional Gated RNN (LCGRNN). they introduced Res-TSSDNet, which is a full-stack model for synthetic speech detection that uses deep feature computation and classification. The model can be modified to fit new data, although this requires more than the usual processing.

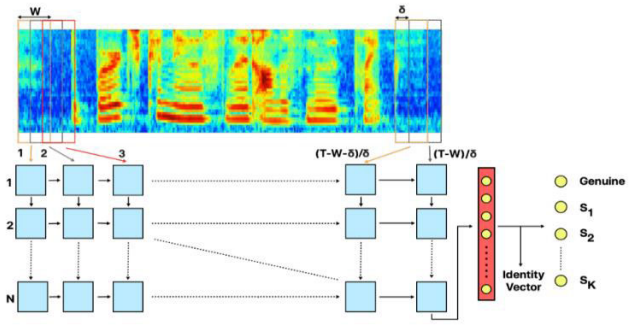


FIGURE 7. The proposed LC-GRNN utterancelevel identity vector extractor block diagram [149].

The initial stage of data preparation involves the elimination of noise and the normalization of features. The process of feature extraction involves the utilization of a (CNN) for the purpose of extracting features from the speech signal. The process of classification employs a (RNN) to differentiate between authentic and fraudulent speech signals.

The following equation effectively encapsulates the temporal dependencies and contextual information present within the audio sequence, thereby playing a pivotal role in the precise detection of counterfeit audio. The equation for updating the Gated Recurrent Unit (GRU) can be expressed as follows:

$$\mathbf{z}_t^n = \sigma\left(\mathrm{MFM}\left(W_z^n * x_t^n + \mathbf{U}_z^n * \mathbf{h}_{t-1}^n\right)\right) \tag{20}$$

The symbol $*$ is utilized to represent a convolution operation performed by an operator. The convolutional layers may be construed as filter banks that have undergone training and optimization to identify anomalies in the counterfeit speech. The primary benefit of utilizing these filters lies in the extraction of frame-level characteristics at each temporal interval, which exhibit greater discriminatory power than those obtained through the utilization of fully connected units.

Also The variable z_t^n denotes the update gate at time t, which governs the extent to which the prior hidden state h_{t-1}^h is modified in response to the present input x_n^t. The computation of the update gate involves the utilization of a sigmoid activation function (σ), in conjunction with the weight matrices W_z^n and U_z^n, and the bias term b_z.

Cheng et al. [150] proposed a strategy that utilized the Squeeze-Excitation Network (SENet) to train a Deep Neural Network (DNN) by incorporating log power magnitude spectra and CQCC acoustic features. The ASVspoof 2019 dataset was utilized to evaluate the method, which demonstrated a 17% enhancement in the identification of synthetic audio. Nonetheless, the model's efficacy exhibited a decline when subjected to a logical access scenario, wherein overfitting was detected, leading to a t-DCF cost and an EER of zero.

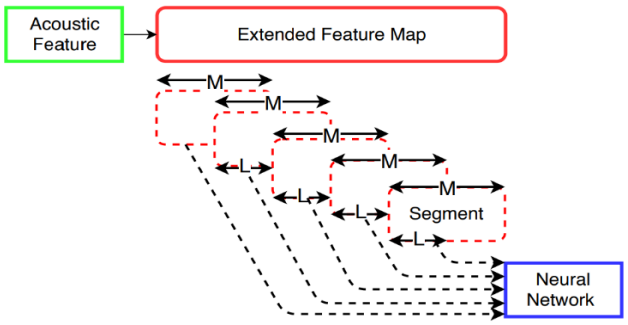


FIGURE 8. Feature map technique using the Unified method illustrated. A unified feature map is created by repeatedly repeating an utterance after extracting low-level acoustical features. The feature map is then divided into segments with M frames of length and L frames of overlap, and input into the DNN models [150].

Figure (8) depicts Feature map technique of the system designed for detecting audio-visual spoofing. The process entails the retrieval of auditory and visual characteristics from the input, which are subsequently merged to capture their interrelation. The temporal modeling module is designed to capture temporal dependencies and contextual information within the fused features. Ultimately, the authenticity of the input is determined through a classification stage that leverages the output generated by the temporal modeling module. The presented block diagram illustrates the integration of auditory and visual data within the system, which serves to augment its ability to detect instances of spoofing.

Alzantot et al. [151] have proposed a technique that utilizes a residual (CNN) for the identification of audio deepfakes. The Counter Major (CM) score of the counterfeit audio is calculated through a technique that involves the extraction of significant features from the input, such as the Mel-Frequency Cepstral Coefficients (MFCC), Constant-Q Cepstral Coefficients (CQCC), and (STFT). The findings indicate a notable enhancement of 71% and 75% in the t-DCF (0.1569) and EER (6.02) matrices, respectively. Nevertheless, the system exhibits generalization errors, underscoring the necessity for additional research to augment its efficacy.

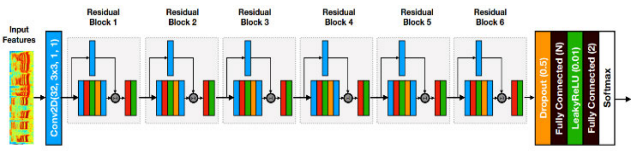


FIGURE 9. The Spec-ResNet model architecture [151].

The system comprises four main parts: pre-processing, feature extraction, classification, and post-processing.

The pre-processing stage removes noise from the audio signal and normalizes it to a standard range. The feature extraction stage extracts feature from the audio signal, such as Mel-frequency cepstral coefficients (MFCCs). The classification stage uses a deep residual neural network (ResNet) to classify the audio signal as authentic or spoofed. The post-processing stage produces the classification decision from the ResNet.

The authors propose investigating alternative methods for feature extraction and incorporating supplementary data into the model to reduce generalization errors.

$$CM(s) = \log(p(\text{bona fide} \mid s; \theta)) - \log(p(\text{spoof} \mid s; \theta)) \tag{21}$$

The equation involves the assignment of probabilities to the input being either genuine or a spoof, denoted as P(genuine) and P(spoof), respectively. The computation of the log-likelihood ratio involves the natural logarithm of the ratio between the probability of genuine events and the probability of spoof events.

Through utilization of the aforementioned formula, the system is capable of producing a numerical value indicative of the probability that the input is authentic or fraudulent. Elevated CM values are indicative of a greater probability of the input's authenticity, whereas reduced values suggest a higher probability of it being a counterfeit.

Rahul et al. [132] introduced a novel methodology for identifying falsified English-speaking audios using transfer learning and the ResNet-34 technique, which outperformed unimodal and multimodal approaches. The CNN network was used for pre-training the transfer learning model, which utilized the Rest-34 technique to address the vanishing gradient problem. The framework yielded optimal outcomes, as evidenced by an EER metric of 5.32% and a t-DCF metric of 0.11514%. Khochare et al. [61] conducted a study on the detection of artificially generated fraudulent audio by utilizing two innovative deep learning models, namely

the Temporal Convolutional Network (TCN) and the Spatial Transformer Network (STN). The study explored the efficacy of feature-based and image-based approaches. The TCN model demonstrated a favorable outcome with an accuracy rate of 92%. In contrast, the STN model exhibited an accuracy rate of 80%; however, it lacked the capability to accommodate inputs such as (STFT) and Mel Frequency Cepstral Coefficient (MFCC).

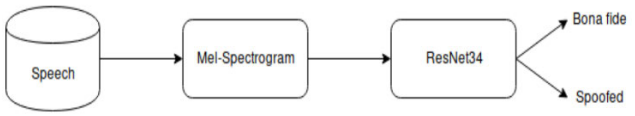


FIGURE 10. The framework of the proposed speech spoofing detection system [132].

The proposed framework uses transfer learning to train a deep CNN to classify speech signals as genuine or fraudulent. The CNN is first trained on a large dataset of speech signals using a pre-existing model. This allows the network to learn universal speech characteristics, which can then be used to classify new speech signals. The CNN is then fine-tuned using a smaller dataset of speech signals that have been labeled as genuine or fraudulent. This fine-tuning allows the neural network to learn the specific characteristics of the speech signals in the dataset, which improves the accuracy of classification.

Mel-spectrograms are used to represent speech signals in the frequency domain. This representation captures both the temporal and spectral characteristics of the signal, which is important for classifying speech signals.

ResNet is a CNN that is known for its ability to learn long-range dependencies in data. This makes it well-suited for classifying speech signals, which can have long temporal dependencies.

The proposed framework has been shown to achieve high accuracy in classification across a variety of speech datasets. This is due to the use of transfer learning, which allows the neural network to learn the fundamental characteristics of speech, and the use of Mel-spectrograms, which captures the temporal and frequency aspects of the signal.

Chintha et al. [57]proposed two novel models for audio deepfake detection. The first model, CRNN-Spoof, uses a bidirectional LSTM network to predict counterfeit audio based on five layers of extracted audio signals. The second model, WIRE-Net-Spoof, uses a weighted negative log-likelihood function and outperformed CRNN-Spoof by 0.132% in the Tandem Decision Cost Function (t-DCF) with an EER of 4.27% in the ASV Spoof Challenge 2019 dataset [152].

Figure (11) depicts the block diagram of the system. The system consists of an audio/video encoder, a feature extractor, and a recurrent neural network (RCNN). The audio/video encoder converts the input signal into a digital format. The feature extractor extracts feature from the digital signal. The RCNN is a machine learning model that is trained to classify the signal as genuine or counterfeit. The decision maker uses the output of the RCNN to determine the authenticity of the signal.

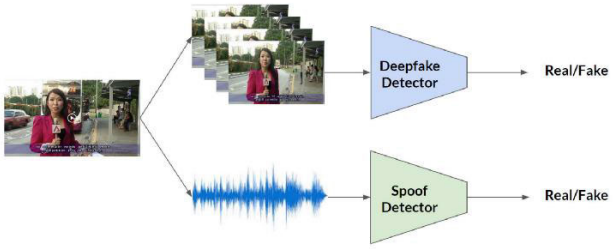


FIGURE 11. An overview of the problem domain. The audio and visual components are extracted and subjected to spoof and deepfake detection models for processing [57].

The present architectural design incorporates mathematical equations, which are outlined below.

$$L_{CE} = -\log\left(\frac{e^{y_c}}{\sum_{j=1}^{1+n_f} e^{y_j}}\right) \tag{22}$$

The cross-entropy loss function (L_CE) is used to train the RCNN. It measures the difference between the predicted probability distribution and the actual probability distribution.

$$\begin{aligned} &D_{KL}(\mathcal{N}((\mu_1, \mu_2)^T, \text{diag}(\sigma_1^2, \sigma_2^2)) \parallel \mathcal{N}(0, I)) \\ &= \lambda(\sum_{i=1}^{n} \sigma_i^2 + \mu_i^2 - \log(\sigma_i) - 1) \end{aligned} \tag{23}$$

The Kullback-Leibler (KL) divergence (L_KL) is used to measure the difference between two probability distributions.

$$L_{EN} = \lambda_1 L_{KL} + \lambda_2 L_{CE} \tag{24}$$

The ensemble loss (L_EN) is a combination of L_CE and L_KL. It is minimized to improve the overall performance of the RCNN.

Shan and Tsai [153] developed a method for aligning audio recordings using three different classification models: LSTM, bidirectional LSTM, and transformer architectures. The goal of this approach was to classify individual audio frames from a set of 50 distinct recordings into either a matching or non-matching status. The bidirectional LSTM model was found to have the best performance, achieving a precision rate of 99.7% and an error rate of 0.43%.

As shown in figure (12) The system consists of three main components:

A repository of unprocessed audio recordings sourced from reliable entities, a search sub-system that retrieves relevant matches from a database in response to an audio query, and a cross-verification sub-system that uses a reference recording to authenticate the audio query and ensure its validity.

The cross-verification sub-system comprises three steps:

First, feature extraction: The audio query and the reference recording are transformed into a collection of features

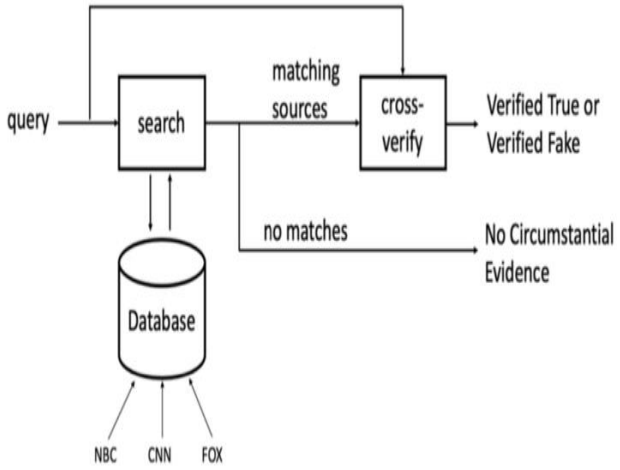


FIGURE 12. An overview of the entire system. The objective is to authenticate a speech recording provided by a prominent global figure [153].

that capture the essence of their acoustic content. Second, Alignment: The features of the audio query and the reference recording are aligned using dynamic programming. Last, decision-making: The system determines the legitimacy of the audio query based on the results of the alignment.

The process of feature extraction involves the computation of Mel-frequency cepstral coefficients (MFCCs) from the audio recordings. The MFCCs comprise a set of 39 dimensions, which encompass both delta and delta-delta features.

The alignment process involves the computation of a pairwise cost matrix C, which is obtained by evaluating the Euclidean distance between the query and reference features. The cumulative cost matrix D is generated through dynamic programming using the following guidelines:

$$D[i,j] = \begin{cases} C[i,j] & i=0 \\ \alpha + D[i-1,j] & i>0, j=0 \\ min(\gamma + D[i,j-1], \alpha + D[i-1,j], \\ C[i,j] + D[i-1,j-1]) & i>0, j>0 \end{cases} \quad (25)$$

The backtrace matrix B is concurrently updated with D in order to maintain a record of the optimal transition types. The optimal alignment can be identified by the lowest cost element located in the final row of D. The determination of the optimal subsequence path is achieved through the process of tracking the back pointers.

The aforementioned equations encapsulate the procedure of aligning the features of a query and a reference through the calculation of costs, cumulative costs, and optimal transitions via dynamic programming.

Wijethunga et al. [154] proposed a system for detecting audio produced by AI synthesizers using a combination of CNNs and RNNs. The system first preprocesses the audio data by converting the sample rate, merging audio channels, and extracting MFCCs. The MFCCs are then used to train a DNN to predict the existence of background noises. The DNN is then used to filter out the background noises from the original audio signal.

The system was evaluated on the UrbanSound8K dataset, which consists of labeled urban audio excerpts from 10 distinct classes. The system achieved a success rate of 94% in detecting audio produced by AI synthesizers.

The system's block diagram is shown in Figure 13. The system consists of three main components,

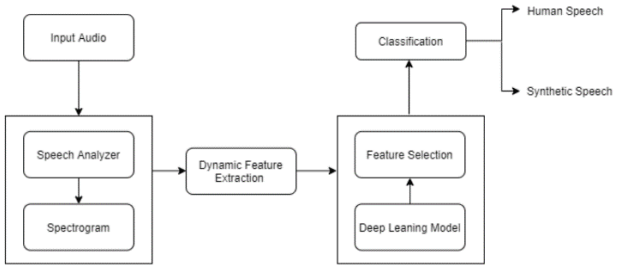


FIGURE 13. Block diagram for synthetic speech detection with DNN [154].

A data preprocessing component that converts the sample rate, merges audio channels, and extracts MFCCs.

A DNN classifier that predicts the existence of background noises.

An adaptive filter that eliminates the background noises from the original audio signal.

The system was able to achieve high accuracy by combining the strengths of CNNs and RNNs. CNNs are good at extracting features from the input data, while RNNs are good at capturing long-term dependencies. By combining these two types of neural networks, the system was able to learn to identify the subtle differences between real and synthetic audios.

The system is a promising step towards developing effective methods for detecting audio produced by AI synthesizers. It can be used to protect against the spread of misinformation and disinformation, and it can also be used to improve the security of audio-based authentication systems.

Jiang et al. [155] introduced a self-supervised spoofing audio detection (SSAD) model, which draws inspiration from PASE+, a pre-existing self-supervised deep learning methodology. The employed approach involves the utilization of multilayer convolutional blocks to extract contextual features from the audio stream. Although SSAD exhibited commendable scalability and efficiency, its performance was comparatively weaker than other deep learning methodologies, as evidenced by an EER of 5.31 percent. Subsequent investigations may delve into the prospective advantages of self-supervised learning and scrutinize techniques to augment its efficacy, with the aim of further ameliorating the SSAD model's performance. Additionally, research could be conducted to investigate the potential of combining self-supervised learning with other DL approaches to create

a hybrid model that could potentially outperform existing models. The features of the MLmodels can be extracted automatically, reducing the need for extensive preprocessing and saving time. To further improve the performance of the models.

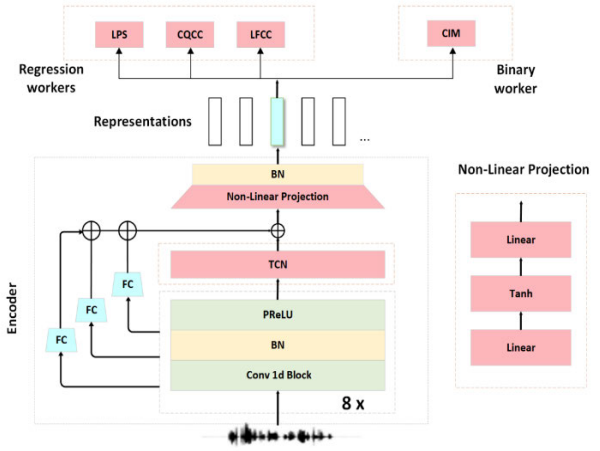


FIGURE 14. The architecture of SSAD [155].

As can be seen from Figure (14) The architecture of SSAD as follow.

SSAD's architecture includes an encoder—a CNN—to extract audio features. The encoder comprises convolutional layers followed by max pooling, extracting features at various scales and reducing dimensionality. Workers, compact neural networks, perform self-supervised tasks on encoder-extracted features to enhance discriminative qualities between real and fake audio. The classifier, another neural network, is trained to categorize recordings based on encoder features and worker predictions, using a dataset of labeled authentic and synthetic audio. The system excels in precise classification due to its self-supervised learning, acquiring distinctive features for differentiation. This learning method outperforms conventional supervised learning that relies on annotated data. SSAD modifies the encoder's architecture accordingly.

The architecture of the encoder is modified by SSAD in the following manner.

The dilated convolution, denoted as F, is an operation performed on an element s within a sequence. It involves the expansion of the receptive field of the convolutional kernel by inserting gaps between the kernel elements. This results in a larger effective kernel size, which allows for the incorporation of a larger context into the convolution operation.

$$F(s) = (x \cdot_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \tag{26}$$

The dilation factor is represented by d, the filter size is denoted by k, and the term $s - d \cdot i$ takes into consideration the past direction. The process of dilation can be understood as the incorporation of a constant interval between each pair of neighboring filter taps.

The quality of the preceding layer's representations is enhanced by the inclusion of an additional hidden layer with ReLU activation in the nonlinear projection.

The Congener Info Max (CIM) task aims to reduce the disparity between two comparable types of speeches, as defined by L1, while simultaneously increasing the disparity between two distinct types of speeches, as defined by L2.

$$L1 = E_{S_r}\left[\log\left(d\left(s_a, s_r\right)\right)\right] \tag{27}$$

$$L2 = E_{S_f}\left[\log\left(1 - d\left(s_a, s_f\right)\right)\right] \tag{28}$$

$$L = L1 + L2 \tag{29}$$

The discriminator function, denoted as d, is evaluated with the expectation over positive samples (E_{S_f}) and negative samples (E_{S_r}).

The A-Softmax loss function, also known as Angular softmax, is a mathematical function used in ML.

The A-Softmax loss function can be denoted as follows:

$$L_{ang} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|X_i\|\cos\left(m\theta_{y_i,i}\right)}}{e^{\|X_i\|\cos\left(m\theta_{y_i,i}\right)} + \sum_{j\neq y_i} e^{\|X_i\|\cos\left(\theta_{y_j,i}\right)}}\right) \tag{30}$$

In the context of ML, the variable N represents the quantity of training samples denoted by the set $\{X_i\}N_{i=1}$, along with their respective labels $\{yi\}N_{i=1}$. These training pairs are utilized in the calculation of $\theta_{y_i,i}$, which represents the angle between X_i and the corresponding column y_i of weights W in the fully connected classification layer. Additionally, the integer m serves as a parameter that governs the magnitude of the angular margin between classes.

Subramani and Rao [156] propose a number of methods for improving the accuracy of fake speech detection models, including, Lightweight convolutional neural networks: The authors propose two lightweight convolutional neural network architectures for fake speech detection: EfficientCNN and RES-EfficientCNN. These models have fewer parameters and require less memory than traditional methods, making them more efficient and easier to deploy on resource-constrained devices.

Multi-task learning: The authors also propose a multi-task learning setting for fake speech detection. In this setting, the model is trained to jointly predict the veracity (bonafide vs. fake) and the source of the fake speech. The authors argue that this helps the model to learn more discriminative features for fake speech detection.

Transfer learning: The authors also investigate the use of transfer learning for fake speech detection. Transfer learning is a technique where a model trained on one task is used as a starting point for training a model on a new task. The authors show that transfer learning can be used to adapt fake speech detection models to new attack vectors (synthesis models) with less training data.

The authors evaluate their methods on two datasets of fake speech: the ASVSpoof2019 dataset [72] and the RTVCSpoof

dataset. They show that their methods significantly outperform previous methods on both datasets.

The findings of the evaluation indicate that RES-EfficientCNN outperformed EfficientCNN with an F1-score of 97.61 points, surpassing the latter's F1-score of 94.14 points by 3.47 points. The aforementioned results indicate that the proposed approach is efficacious in enhancing the precision of the model.

Lei et al. [157] proposed a hybrid model that integrates 1-D CNN and Siamese CNN to optimize the performance of the latter. The hybrid architecture was formulated by amalgamating two CNN and appending a fully connected layer at the terminal stage. The results obtained from the experiment indicate that the employment of the hybrid model led to a notable enhancement of around 50% in both the min-tDCF and EER metrics, specifically when utilizing the LFCC features. The utilization of CQCC features in conjunction with the hybrid model resulted in a notable enhancement in model performance, as evidenced by a roughly 20% improvement in both the min-tDCF and EER metrics. The results of this study indicate that the hybrid model exhibits greater resilience and efficacy in accommodating diverse feature sets. Furthermore, the hybrid model exhibited greater resilience to noise and improved capacity for detecting fraudulent audio.

The system is composed of two fundamental components, specifically a feature extractor and a classifier.

Feature extractor, as shown in figure (15) This component converts the raw audio signal into a set of discrete features that can be used by the classifier to differentiate between genuine and deceptive speech. The feature extractor used in this system is a one-dimensional convolutional neural network (1)-D CNN). The CNN is provided with a set of log-probabilities, which have been generated for each frame of the audio signal through the utilization of a Gaussian mixture model (GMM) as its input. Following this, the 1-D CNN produces a set of features that illustrate the local and global relationships between the frames.

Classifier, this component receives a sequence of features extracted by the feature extractor and produces a probability estimate of the authenticity of the speech signal. The classifier used in this system is a Siamese CNN as can be seen from Figure (16). The Siamese CNN is composed of two identical CNNs that undergo simultaneous training on identical datasets. The two CNNs have identical weights and biases, albeit having distinct inputs and outputs. The two CNNs receive inputs in the form of feature sequences extracted from two distinct utterances. The probabilities indicating the authenticity of the two utterances are generated as outputs by the two CNNs. The Siamese CNN integrates the two probabilities to generate a conclusive probability regarding the authenticity of the speech in the given input utterance.

$$\mathrm{p(x)} = \sum_{i=1}^{M} w_i p_i(x) \tag{31}$$

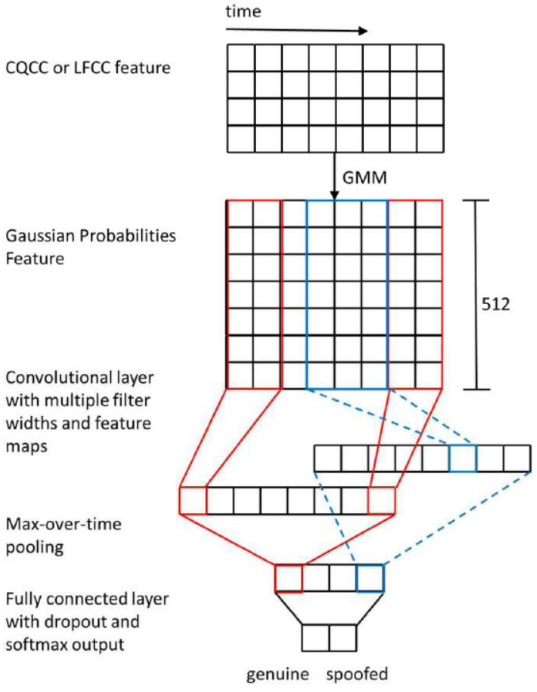


FIGURE 15. The architecture of the CNN model [157].

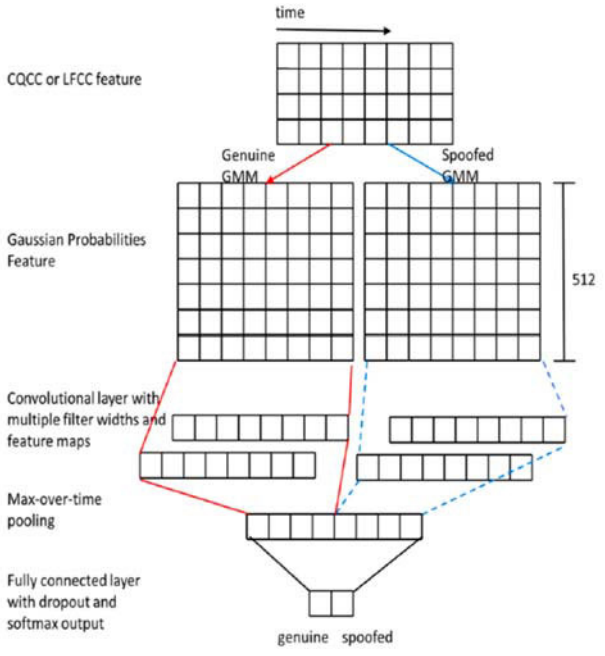


FIGURE 16. The architecture of the Siamese Convolutional Neural Network (CNN) model [157].

The log-probabilities of each frame in the audio signal are computed utilizing the (GMM). The log-probabilities denote the probability that the frame was produced by an authentic speaker or an impostor speaker.

$$\mathrm{p}_i(\mathrm{x}) = \frac{1}{(2\pi)^{D/2}\,|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}\,(x-\mu_i)'\,\Sigma_i^{-1}\,(x-\mu_i)\right\} \tag{32}$$

1-D (CNN) is a neural network architecture that is frequently employed in the domains of speech and image processing. The (CNN) implemented in this system comprises 512 filters.

$$\text{score}_{\text{baseline}} = \log p\left(X \mid \lambda_h\right) - \log p\left(X \mid \lambda_s\right) \tag{33}$$

The Siamese (CNN) is a prevalent neural network architecture utilized for various applications, including but not limited to object matching and facial recognition. The system employs a Siamese (CNN) architecture, wherein two CNNs with identical structures are concurrently trained on the same dataset.

$$\mathrm{f}_{ij} = \log\left(w_j \cdot p_j\left(x_i\right)\right) \tag{34}$$

In the context of speech feature sequences, it is observed that the GMM approach operates by independently accumulating scores across all frames, without taking into account the specific contribution of each Gaussian component towards the final score.

Furthermore, the disregard for the correlation between consecutive frames has been observed. The objective is to construct a model for the distribution of scores on each component of the (GMM) and introduce the Gaussian probability feature.

In this experiment, it was observed that for a raw frame feature such as CQCC or LFCC, the size of the new feature f is dependent on the order of GMM. Additionally, the component f7 is also a crucial factor.

Subsequently, the mean and standard deviation values of the training dataset are computed and subsequently employed for the purpose of mean and variance normalization for every individual utterance.

Lataifeh et al. [158] conducted an experimental study aimed to evaluate the effectiveness of ML(ML) models in comparison to (CNNs) and Bidirectional Long Short-Term Memory (BiLSTM) in detecting imitation-based fakeness on the Arabic Diversified Audio (AR-DAD) dataset [159]. The research conducted an investigation on a range of MLmethodologies, encompassing SVM, SVM-Linear, Radial Basis Function (SVMRBF), LR, Decision Tree (DT), Radial Basis Function (RF), and Gradient Boosting (XGBoost). According to the findings, the Support Vector Machine (SVM) exhibited the most noteworthy precision rate of 99%, whereas the Decision Tree (DT) demonstrated the least accuracy rate of 73.33%. CNN attained a detection rate of 94.33%, surpassing the performance of BiLSTM. CNN demonstrated a high level of efficacy in detecting false correlations and autonomously extracting characteristics through its capacity for generalization. Nonetheless, a limitation of (CNN) architectures in the context of Audio Deepfake pertains to their exclusive capacity to handle visual data as input. Preprocessing of the audio is necessary to convert it into a spectrogram or a two-dimensional representation prior to input into the network.

Classifier systems based on MLtechniques are employed to classify data by utilizing input features. The classifiers can be

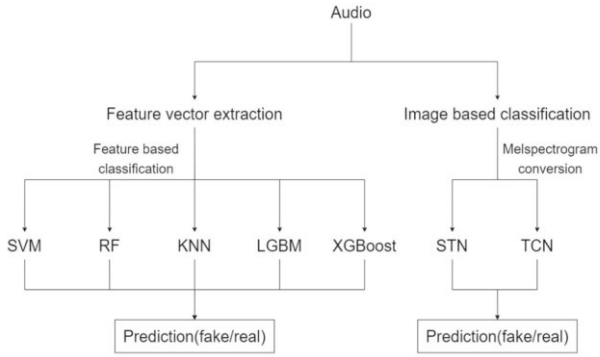


FIGURE 17. The representation of all classifiers that have been implemented in their entirety [158].

categorized into two domains, namely shallow learning and deep learning. Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), and Gradient Boosting: XGBoost (XG) are examples of shallow learning classifiers. The category of deep learning classifiers encompasses Convolution Neural Networks (CNN) and Bidirectional (BiLSTM).

Khochare et al. [61] conducted a comprehensive investigation to evaluate the effectiveness of feature-based and image-based techniques in the classification of synthetically produced counterfeit audio. The present study employed two innovative deep learning models, namely the Temporal Convolutional Network (TCN) and the Spatial Transformer Network (STN), to achieve the intended objective. The findings of the study indicate that TCN exhibited a high level of precision in distinguishing authentic from fabricated audio, with a notable accuracy rate of 92%. In contrast, STN demonstrated a comparatively lower accuracy rate of 80%. Despite exhibiting exceptional performance with sequential data, it was discovered that the (STFT) and (MFCC) features, when transformed into inputs, were incompatible with TCN, as per the findings.

As shown in figure (18), The system proposed in this study consists of two approaches, feature-based classification and image-based classification.

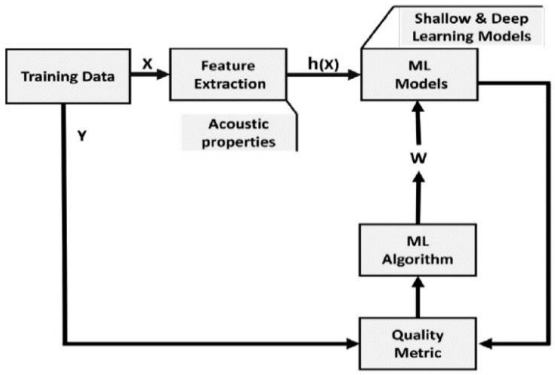


FIGURE 18. Diagram of the method for detecting deepfake audio [61].

Feature-based classification: This approach converts audio samples into a dataset of features, such as mean square energy, Chroma features, spectral centroid, spectral bandwidth, spectral roll off, zero crossing rate, and MFCCs. These features are then used to train machine learning (ML) models, such as support vector machines (SVMs), light gradient boosting machines (LGBMs), extreme gradient boosting (XGBoosts), k-nearest neighbors (KNNs), and random forests (RFs). The trained models are then used to classify new audio samples as either authentic or counterfeit.

Image-based classification: This approach converts audio samples into melspectrograms using the librosa library. Melspectrograms are visual representations of the frequency content of audio signals. The melspectrograms are then used to train deep learning models, such as spatial transformer networks (STNs) and temporal convolutional networks (TCNs). The trained models are then used to classify new audio samples as either authentic or counterfeit.

The concept of Mean Square Energy of a signal x(n) can be expressed as follows:

$$x_{\text{rms}} = \sqrt{\frac{1}{n}\left(x_1^2 + x_2^2 + \cdots + x_n^2\right)} \tag{35}$$

In this context, the variable ''n'' represents the total number of samples, while $x_i = i$th sample.

Spectral centroid: The spectral centroid is a measure of the center of gravity of the spectrum of a signal. It is calculated as the weighted average of the frequencies in the spectrum, with the weights being the magnitudes of the frequencies.

$$\mu = \frac{\sum_{k=1}^{k=N} f(k) \cdot f(k)}{\sum_{k=1}^{n=N} m(k)} \tag{36}$$

The magnitude at the kth frequency bin is denoted as m(k), while the center frequency at the kth frequency bin is represented by f(k).

Spectral bandwidth: The spectral bandwidth is a measure of the width of the spectrum of a signal. It is calculated as the square root of the variance of the frequencies in the spectrum.

$$\left(\sum_k m(k)(f(k) - \mu)^p\right)^{\frac{1}{p}} \tag{37}$$

The expression $m(k)$ denotes the magnitude at the kth frequency bin, while $f(k)$ represents the center frequency at the same bin. The parameter μ corresponds to the spectral centroid.

Spectral rolloff: The spectral rolloff is a measure of the frequency below which a certain percentage of the total energy in the spectrum is located. It is calculated as the frequency at which 85% of the energy in the spectrum is located.

$$\arg \max_{f_r \in \{1,\ldots,N\}} \sum_{k=1}^{f_r} m(k) \geq 0.85 \sum_{k=1}^{N} m(k) \tag{38}$$

The rolloff frequency is denoted as f_r, and the magnitude at the kth frequency bin is represented by $m(k)$.

The computation of the Zero Crossing Rate:

Zero crossing rate: The zero crossing rate is a measure of the frequency at which a signal crosses the zero axis. It is calculated as the number of times the signal crosses the zero axis in a given time interval.

$$\frac{1}{W_L} \sum_{n=1}^{W_L} |\text{sgn}[x(n)] - \text{sgn}[x(n-1)]| \tag{39}$$

The given expression pertains to an audio signal represented by $x(n)$, wherein W_L denotes the window length and sgn represents the signum function.

In their study [160], E.R. Bartusiak and E.J. Delp proposed a novel approach to assign synthetic speech to its originator. The employed technique utilizes a transformer, which is a neural network framework that has demonstrated efficacy in various natural language processing endeavors. The efficacy of the method was evaluated on three distinct sets of synthetic speech data, and it demonstrated a notable level of precision across all three datasets. The method attained a 99.8% accuracy rate on the ASVspoof2019 dataset.

The method attained a 96.3% accuracy on the SP Cup dataset. The method attained a precision rate of 93.4% on the DARPA SemaFor Audio Attribution dataset. The efficacy of the technique was also evaluated in an open-set context, wherein it demonstrated the ability to accurately detect unfamiliar speech generation techniques, achieving a precision rate of 90.2% on the ASVspoof2019 dataset and 88.45% on the DARPA SemaFor Audio Attribution dataset.

The method exhibits robustness towards AAC compression when the data rates are equal to or greater than 32kbps. The method's transformer comprises a total of approximately 87 million parameters. The authors intend to enhance the precision and resilience of the approach in their forthcoming research.

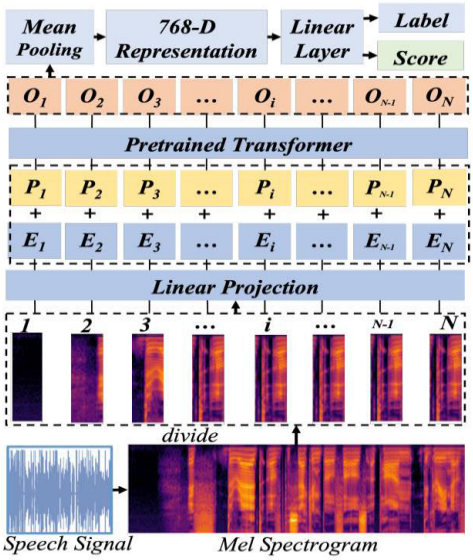


FIGURE 19. The diagrammatic representation of the proposed approach, namely Synthetic Speech Attribution Transformer (SSAT) [160].

In Figure (19), the initial stage transforms speech into a melspectrogram, emphasizing frequencies significant for human hearing. It partitions the spectrum into mel bands and

computes power spectra within each. Mel bands are logarithmic, enhancing perceptual significance. The melspectrogram is segmented for better classification precision, providing additional data to the classifier. Each region is assigned a vector with statistical speech signal characteristics. Positional encoding is then added to each vector for the transformer neural network to understand their relative positions. The transformer network excels at capturing distant relationships using self-attention.

The transformer processes vectors and encodings, generating concealed states. These states are aggregated to create a single 768-dimensional representation for the auditory input. Categorization is done through a linear layer with SoftMax activation. It transforms the 768-dimensional representation into a probability distribution over potential classes, ensuring their total equals unity. The output includes a classification label denoting speech origin and a confidence score reflecting the classifier's certainty.

However, ASVspoof 2021 [161] presents new challenges. It introduces a category of compressed TTS and VC deepfake samples without speaker verification or original speakers' audios.

Arif et al. [162] introduced a new audio feature descriptor, named ELTP-LFCC, which is created by merging two existing techniques: Local Ternary Pattern (ELTP) and Linear Frequency Cepstral Coefficients (LFCC).

The researchers utilized a Deep Bidirectional Long Short-Term Memory (DBiLSTM) network in conjunction with this descriptor to construct a model capable of detecting fraudulent audio in diverse indoor and outdoor settings. The ASVspoof 2019 dataset, comprising of artificially generated and impersonation-based fraudulent audio, was utilized to assess the efficacy of the model. The findings indicate that the model exhibited greater efficacy in identifying artificially generated audio (with an equal error rate of 0.74%) as compared to samples produced through imitation (with an equal error rate of 33.28%).

The block diagram can be explained as shown in Figure (20), where a bidirectional LSTM model classified using ELTP-LFCC features. Each BiLSTM layer had 64 units. Concatenated outputs were passed to a FC layer, then a softmax layer for classification.

The suggested architecture integrated ELTP, LFCC, and BiLSTM to accurately detect logical access attacks in audio signals.

Extended Local Tertiary Patterns (ELTP):

$$P\left(s^i, \mathrm{c}, \theta\right) = \begin{cases} 1, & s^i \geq c + \theta \\ 0, & \left|\left(s^i - \theta\right)\right| < \theta \\ -1, & s^i \leq (\mathrm{c} - \theta) \end{cases} \tag{40}$$

The acoustic signal is denoted by $P\left(s^i, \mathrm{c}, \theta\right)$, where c corresponds to the central sample of the frame F that has s^i neighbors. The neighbor index is represented by i, while the threshold is denoted by θ. The ELTP is computed by determining the magnitude difference between the central

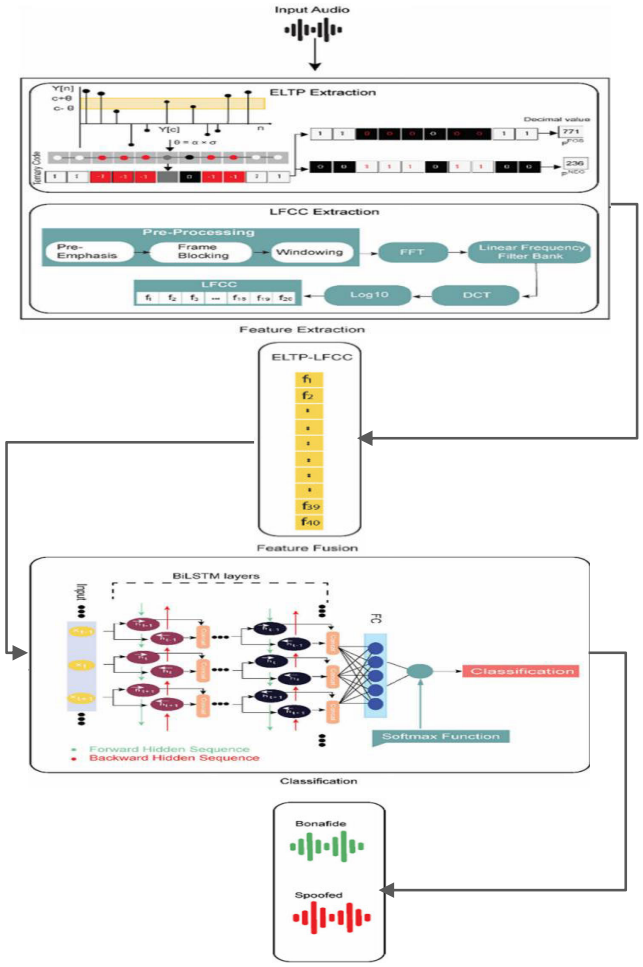


FIGURE 20. The proposed framework's architectural design [162].

sample c and the 10 adjacent audio samples i, through the application of θ around c.

The auto-adaptive threshold is computed dynamically by utilizing the standard deviation of each frame.

$$\theta = \alpha \times \sigma(0 < \alpha \leq 1) \tag{41}$$

The symbol σ denotes the standard deviation that is calculated for every frame of the audio, while α represents a scaling factor. A linear search algorithm was utilized to optimize the scaling factor α by identifying the point of convergence within the interval of 0 and 1.

Linear Frequency Cepstral Coefficients (LFCC)

The computation of the 20-dimensional Linear Frequency Cepstral Coefficients (LFCC) involves the utilization of a series of linear filters on the Fast Fourier Transform (FFT) of the audio signals.

$$\sum_{k=1}^{K} \log\left(g_k\right) \cos\left(\frac{(2k-1)\, i\pi}{2K}\right), \quad 1 \leq i \leq I \tag{42}$$

In the given context, K denotes the quantity of filters while I represent the number of Local Feature Coding Coefficients (LFCC) utilized. The final 40-dimensional ELTP-LFCC

feature vector was obtained by integrating the 20-dimensional LFCC features with the 20-dimensional ELTP features.

Bidirectional Long-Term Short-Term Memory (BiLSTM):

BiLSTM's calculation of the concealed vector and the output vector

$$h_t = H\left(W_{xh}x_t + W_{hh}h_{t-1} + B_h\right) \tag{43}$$

$$y_t = W_{hy}h_t + b_y \tag{44}$$

The variables in the equation are denoted as follows: *W*represents the weight matrices, where W_{xh} specifically denotes the input-hidden weight matrix. B represents the bias vectors, with B_h representing the hidden bias vector. Finally, H represents the hidden function.

The computation involved in the Long Short-Term Memory (LSTM) cell pertains to the forget gate, input gate, output gate, cell memory, and hidden vector.

$$f_t = \sigma_g\left(W_{xf} \times x_t + W_{hf} \times h_{t-1} + W_{cf} \times c_{t-1} + B_f\right) \tag{45}$$

$$i_t = \sigma_g\left(W_{xi} \times x_t + W_{hi} \times h_{t-1} + W_{ci} \times c_{t-1} + B_i\right) \tag{46}$$

$$o_t = \sigma_g\left(W_{xo} \times x_t + W_{ho} \times h_{t-1} + W_{co} \times c_t + B_o\right) \tag{47}$$

$$c_t = f_t c_{t-1} + i_t \tanh\left(W_{xc} \times x_t + W_{hc} \times h_{t-1} + B_c\right) \tag{48}$$

$$h_t = o_t \tanh\left(c_t\right) \tag{49}$$

The hard-sigmoid function, denoted as σ_g, is utilized in the context of the forget gate (f), input gate (i), output gate (o), cell memory (c), and hidden vector (h).

combining the outputs of the forward and backward hidden sequences.

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\underset{h}{\leftarrow}y}\overleftarrow{h}_t + B_y \tag{50}$$

A sequence that is forward hidden $\vec{h}$, The backward hidden sequence $\overleftarrow{h}$ and output sequence are obtained through an iterative process that involves the forward layer being iterated from t = 1 to T, and the backward layer being iterated from t = T to 1.

The equations presented encapsulate the fundamental principles of the proposed methodology, encompassing both the feature extraction components (ELTP and LFCC) and the classification aspect (BiLSTM) of the framework.

Ballesteros et al. [10] developed a classification model called Deep4SNet that employed a 2D CNN model (histogram) to encode the audio dataset and discriminate between synthetic and imitation audios. This model was incredibly accurate, with an impressive 98.5% accuracy rate when it came to identifying counterfeit and synthetic audio.

Unfortunately, the performance of Deep4SNet was not scalable and was negatively impacted by the process of data translation, thus limiting its potential applications.

As can be seen from Figure (21) The audio data is initially subjected to pre-processing, wherein it is transformed into a histogram image. The aforementioned process involves the segmentation of the audio signal into equidistant temporal intervals, followed by the computation of the frequency count for each interval.

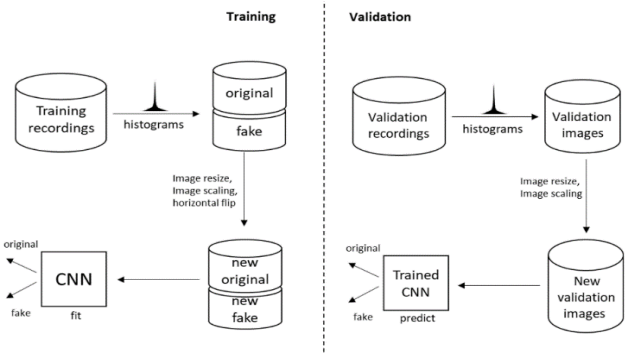


FIGURE 21. Conceptual diagram of the proposed approach [10].

Subsequently, the (CNN) is trained on the histogram images by the system. The (CNN) acquires the ability to discern distinctive attributes within the images that are indicative of counterfeit audio recordings.

Upon completion of the training process, (CNN) can be employed to categorize novel audio data as either authentic or counterfeit.

Binary crossentropy loss function

$$L(y, \hat{y}) = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot \log\left(\hat{y}_i\right) + \left\langle(1 - y_i) \cdot \log\left(1 - \hat{y}_i\right)\right\rangle \tag{51}$$

he selected loss function is binary crossentropy $L(y, \hat{y})$, which is related to the dissimilarity in terms of entropy between two data sequences, in our case, the entropy of the known labels y_i, and the entropy of the predicted labels $\hat{y}$. This kind of loss function is very useful in binary classification problems.

RMSprop optimizer

$$f(x) = max(0, x) \tag{52}$$

The optimizer is utilized for the purpose of training the (CNN). The approach in question pertains to a form of the stochastic gradient descent algorithm that incorporates a rolling average of the squared gradients for the purpose of weight updating in a (CNN).

The Rectified Linear Unit (ReLU) activation function

$$f(x) = \frac{1}{1 + e^{-x}} \tag{53}$$

The (ReLU) activation functions are utilized for the convolutional and hidden layers due to their favorable balance between computational cost and performance. The objective of the (ReLU) is to eliminate negative values while permitting positive values to propagate, as specified by Equation (52).

Sigmoid activation function

$$f(x) = \frac{1}{1 + e^{-x}} \tag{54}$$

The activation function of the final neuron is sigmoid, as derived from Equation (54). The scientific community widely recommends this type of activation for binary classification problems.

The field of Deepfake identification has been expanded by the release of the FakeAVCeleb dataset [163],

Khalid et al. [164] conducted an investigation into the efficacy of unimodal techniques for detecting Deepfakes. Specifically, they evaluated the performance of five classifiers, namely MesoInception-4, Meso-4, Xception, EfficientNet-B0, and VGG16. The research aimed to evaluate the efficacy of unimodal techniques in detecting Deepfakes. The Xception classifier demonstrated the highest level of efficiency, yielding a 76% outcome, whereas the EfficientNet-B0 classifier exhibited the lowest level of performance, producing a result of 50%. Nevertheless, the investigation demonstrated that all unimodal classifiers were unsuccessful in accurately detecting counterfeit audio despite their endeavors.

The model comprises of two distinct components, namely a visual network and an audio network. The (CNN) known as the visual network has been trained to detect visual anomalies present in deepfake videos. The (RNN) known as the audio network has been trained to detect audio artifacts present in deepfake audio.

The operational mechanism of the system commences with the preliminary processing of the input audio and video data. The preprocessing of video data involves the conversion of the data into a series of still images. The audio information undergoes preprocessing through the transformation into a sequence of (MFCC) features. Subsequently, the data that has undergone preprocessing is inputted into the visual and audio networks. The visual network generates a probability score indicating the likelihood of the input video being a deepfake. The audio network generates a probability score indicating the likelihood of the input audio being a deepfake. The ultimate likelihood of the input being a deepfake is determined by the multiplication of the probabilities derived from the visual and audio networks

A (CNN) architecture has been proposed by the authors of a recent academic publication [165], with the intention of addressing the issue of generalization that is frequently experienced in deep learning models. Before the audio data could be fed into the architecture of the CNN, it was first transformed into scatter plot images of adjacent samples. This was done so that it could be used to overcome the challenge. On the Fake or Real (FoR) dataset [147], the accuracy of the model was evaluated, and the results showed that it had a performance of 88.9%. However, its accuracy of 88% and EER of 11% were lower than those of other DL models tested in the study. This indicates the need for additional development as well as the inclusion of more data transformers in order to improve its performance.

Almutairi and Elgibreen [166] Proposed a deep neural network architecture for the purpose of identifying manipulated audio content, commonly referred to as deepfakes. The proposed model was derived from the HuBERT pre-trained model, a substantial language model that underwent training on an extensive corpus of unannotated speech data. The model underwent fine-tuning using a dataset comprising both audio deepfakes and authentic audio recordings. The PyTorch deep learning framework was utilized to implement the system, which underwent training on a dataset consisting of 1000 audio deepfakes and 1000 authentic audio recordings. An assessment was conducted on a corpus comprising 500 fabricated audio files and 500 authentic audio recordings, whereby the system attained a precision rate of 97%.

Figure (22) provides a visual representation of the block diagram for the system. The audio recording is used as an input to the system, which then extracts vector representations from the recording using a feature convolutional layer. The output of the system is the vector representations. After that, the vector representations are separated into audio streams that are either masked or unmasked, depending on which state they are currently in. Encoding processes are applied to the inputs after they have been unmasked. These processes lead to the generation of significant and uninterrupted latent representations. The TE is responsible for deriving the contextualized representations, and it does so by first receiving the encoded inputs that have not been masked and then using those inputs. After obtaining contextualized representations, the next step is to input them into the projection layer, which is responsible for projecting the ultimate context vector. After the context vector has been obtained, it is then sent on to the ASP layer by utilizing the Mean (μ) metric as the transmission method. The result that was produced by

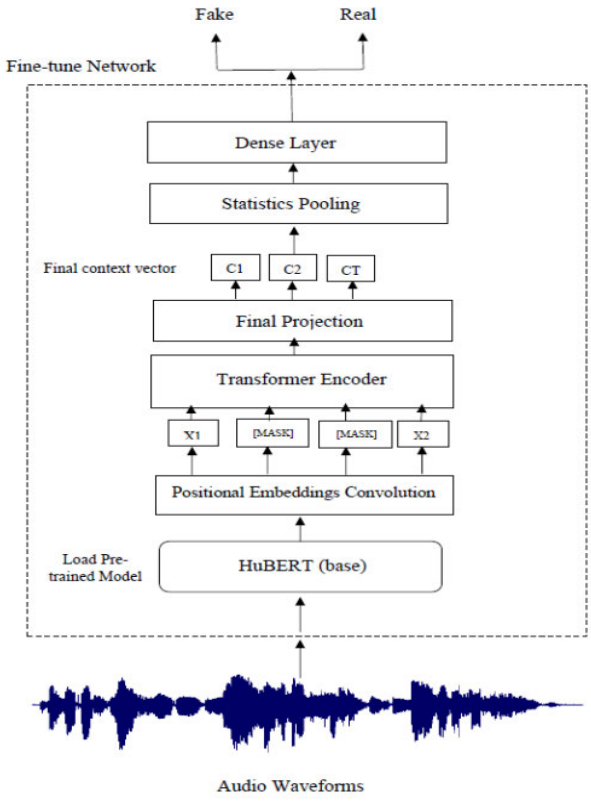


FIGURE 22. The proposed method for detecting Arabic-language Audio Deepfake [166].

TABLE 4. Comparing the effectiveness of classical ML and DL models in detecting fake audio.

Type	Model	Success Rate / Accuracy	Comparison with Other Models
Machine Learning	M Logistic Regression (LR)	98%	N/A
	M (Q-SVM)	97.56%	Outperforms other classical methods
	M Support Vector Machine (SVM)	71%	Higher accuracy than Random Forest (RF)
	M SVM	99%	Highest accuracy among all tested models
	M Decision Tree (DT)	73.33%	Lowest accuracy among all tested models
DL	CNN	99%	More robust than SVM
	BiLSTM	94.33%	Lower accuracy than CNN
	EfficientCNN	94.14 F1-score	Lower F1-score than RES-EfficientCNN
	RES-EfficientCNN	97.61 F1-score	Higher F1-score than EfficientCNN
	Deep4SNet	98.5% accuracy	Affected by data transformation process
	CNN (baseline classifier)	85.99% accuracy	Higher accuracy than random method, but suffers from overfitting

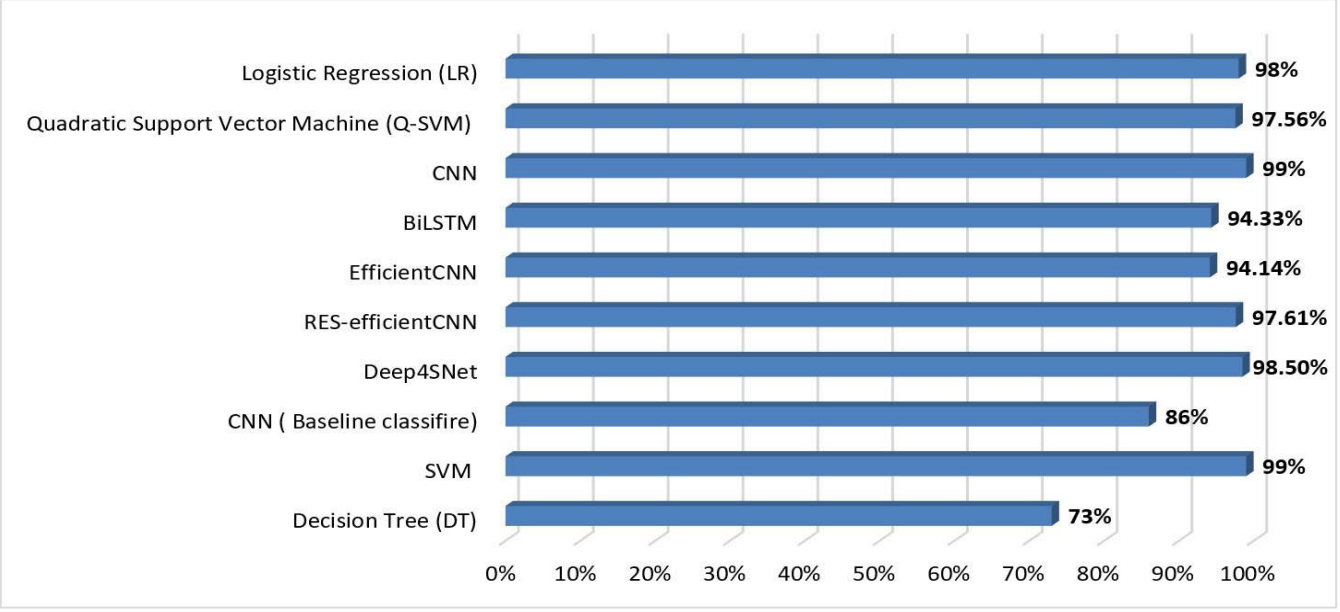


FIGURE 23. The effectiveness of classical machine learning and deep learning models in detecting fake audio.

the ASP layer is then sent onward to a dense layer that makes use of a Tanh activation function. The result that is produced by the dense layer is a forecast regarding the authenticity of the audio recording, more specifically whether or not it is a deepfake.

The Tanh activation function, which is employed in the dense layer is:

$$f(x) = \left(\frac{e^x - e^{-x}}{e^x + e^{-x}}\right) - (1.10) \tag{55}$$

The mathematical constant known as e is the foundation upon which the natural logarithm is built. It is denoted by the variable known as e. The value read from the input device is represented by the variable x.

In the following equation, the cross-entropy loss function is specified. This is a method that is frequently utilized for the training of models.

$$Jbce = -\frac{1}{M}\sum_{m=1}^{M}[ym \times log(h\emptyset(Xm)) + (1 - ym) \times log(1 - h\emptyset(Xm))] \tag{56}$$

The notation M denotes the quantity of training examples in a given dataset. The label of the m training example is represented by ym, while Xm denotes the inputs associated with the m training example. The function $h\emptyset$ is employed to represent the method that utilizes hidden neural network weights $\emptyset$.

Table 4 and Figure 23 indicates that classical MLmodels, namely logistic regression (LR), quadratic support vector

TABLE 5. Comparing the performance of various methods for audio deepfake detection.

STUDY	METHOD	RESULT
Yu et al.[150]	DNN-HLLs	DNN-HLLs: EER 12.24
Cheng et al. [151]	SENet	Relative improvement of 17% in synthetic audio
Alzantot et al. [152]	Residual CNN	Improvement of 71% and 75% in t-DCF and EER, respectively
P. Rahult et al.[138]	Transfer learning and ResNet-34	Best results with EER 5.32% and t-DCF 0.11514%
Khochare et al.[61]	TCN and STN	TCN: 92% accuracy; STN: 80% accuracy
Chintha et al. [57]	CRNN-Spoof and WIRE-Net-Spoof	CRNN-Spoof superior to WIRE-Net-Spoof by 0.132% t-DCF, 4.27% EER
Shan and tsai [154]	LSTM, bidirectional LSTM, transformer	Method assigns matching or non-matching status to audio frames
T. Arif et al.[162]	ELTP-LFCC and DBiLSTM	Synthetic audio: EER 0.74%; Imitated-based samples: EER 33.28%
M. Ballesteros et al.[10]	Deep4SNet	Accuracy rate of 98.5% in identifying counterfeit and synthetic audio
Khalid et al.[164]	MesoInception-4, Meso-4, Xception, EfficientNet-B0, VGG16	Xception: 76%; Efficient Net-B0: 50%
D.camacho et al.[165]	CNN	Accuracy 88.9%; EER 11%

machine (Q-SVM), and SVM, have been employed in several investigations for detecting counterfeit audio, and have demonstrated notable levels of efficacy. Nevertheless, these models frequently necessitate substantial preprocessing and feature extraction. Conversely, (CNNs) and bidirectional long short-term memory (BiLSTM) are alternative deep learning models that have been employed to address the same problem, yielding mixed outcomes. Several studies have reported varying results regarding the comparative robustness of (CNNs) and Support Vector Machines (SVMs). While some studies have demonstrated the superior robustness of CNNs, others have indicated that SVMs exhibit the highest accuracy among all tested models. The selection of a suitable model for detecting fake audio is contingent upon several factors, such as the characteristics and magnitude of the dataset, the intricacy of the features, and the extent of preprocessing necessary.

Overall, it appears that both classical ML and DL models can be effective in detecting fake audio, with the choice of model depending on the specific task and dataset.

Furthermore, the ability to accurately detect fake audio is crucial in maintaining the integrity of information and protecting against malicious intent. In Table 5, we will compare the effectiveness of classical ML and DL, in detecting fake audio, the data will demonstrate the relative performance of each model.

In addition to TABLE 5 the RES-EfficientCNN model, a (CNN) developed by Subramani and Rao [156], achieved a F1-score of 97.61 when tested on the ASV spoof challenge 2019 dataset [152]. The Deep4SNet model, which uses a 2D CNN model to classify imitation and synthetic audio, had an accuracy of 98.5% in detecting such audio. The (SVM) model had the highest accuracy at 99% among the ML models tested by Lataifeh et al. [158] on the Arabic Diversified Audio (AR-DAD) dataset [159], while the decision tree (DT) model had the lowest accuracy at 73.33%. The CNN model had a higher detection rate than the BiLSTM model, with 94.33% accuracy.

The Siamese CNN model proposed by Lei et al. [157] improved the min-tDCF and Equal Error Rate (EER) by approximately 55% when compared to other models, but its performance was slightly lower when using certain features. The CNN model developed in [165] and trained on the Fake or Real (FoR) dataset [147] achieved an accuracy of 88.9% and an EER of 11%.

It is worth mentioning that the performance of these methods may vary depending on the specific dataset and evaluation criteria used. Further research is needed to improve the accuracy and robustness of audio deepfake detection methods.

III. DATASETS

Baidu Dataset, Baidu is a tool for spotting replicated speech, and is a collection created by AI researchers at Baidu's Silicon Valley outpost [70]. There are ten authentic recordings of human speech in this collection, along with 120 cloned samples and four morphed samples.

Mozilla TTS, The world's largest publicly accessible database of speakers has been made available via the widely used open-source browser Mozilla Firefox [167]. As of 2019, the database currently has over 1,400 h of voice recordings in 18 different languages. The audio was recorded in 54 other languages over 7,226 h. These 5.5 million audio samples were used using the Deep Speech Toolkit from Mozilla.

Fake-or-Real (FOR), Another popular dataset utilized in SVR research is the FOR database [147]. Roughly 195,000 snippets of both human and AI-generated speech may be

TABLE 6. Comparing audio Deepfake detection data sets.

DATA SET	RELEASED	TOTAL SIZE	FORMAT	SAMPLES TYPE	LANGUAGES	SPEAKER ACCENT	WEB SITE
BAIDU DATASET [70]	2018	6	mp3	Synthetic	1	English, British & US	https://audiodemos.github.io
MOZILLA TTS [167]	2019	7226	mp3	Imitation Synthetic	54	8% British 24% US	https://commonvoice.mozilla.org/en/datasets
FOR DATASET [142]	2019	-	mp3	Synthetic	1	Native	https://bil.eecs.yorku.ca/datasets
ASV SPOOF 2019 [72]	2019	-	mp3	Synthetic	1	Native	https://datashare.ed.ac.uk/handle/10283/3336
M-AILABS DATASET [168]	2019	18.7 hrs.	WAV	Synthetic	9	Native	https://www.caito.de/2019/01/03/the-m-ailabs-speech-dataset/
AR-DAD: ARABIC DIVERSIFIED AUDIO [159]	2020	16,209 Files	WAV	Imitation	1	Classical Arabic	https://data.mendeley.com/datasets/3kndp5vs6b/
H-VOICE [169]	2020	6672 Files	PNG	Imitation Synthetic	5	English, French, Portuguese, Spanish, and Tagalog	https://data.mendeley.com/datasets/k47yd3m28w/4
FAKEAVCELEB [163]	2021	20,490 Files	MP3	Synthetic	1	English	https://sites.google.com/view/fakeavcelebdash-lab
ADD [170]	2022	85 h	WAV	Synthetic	1	Chinese	*http://addchallenge.cn/add2022*

found in this collection. Human speech samples and examples of recently developed TTS techniques (such as Google Wavene [85] and [126]) are available in this repository. The Authors provide four unique variations of FOR: the "for-original," "for-norm," "for 2 seconds," and "for rec" (FR). Audios in FO are not symmetrical and have not been edited, whereas those in FN are balanced and unchanged. The F2S uses FN data sampled every two seconds to simulate speech-based invasion. By contrast, FR is simply a re-recording of the F2S database.

ASV spoof 2019, One of the most well-known datasets for detecting fake audios [72] is divided into two parts: one for analyzing physical access and the other for analyzing logical access. Both the LA and PA were created using audio samples from 107 unique speakers included in the VCTK basic corpus (46 m and 61 f). LA features examples of converting and cloning audios in addition to both the original and reconstructed audios. The speaker samples aged 20, 10, and 48 years were included in the database. The training, development, and evaluation databases were subdivided based on the two main types of data (21 males and 27 females). All source samples were recorded under the same conditions, notwithstanding the presence of variable presenter categorizations. The evaluation set contained examples of unknown attacks, whereas spoofing cases of the same type and parameters were included in the development and training sets, respectively.

M-AILabs, The real-speech dataset created by M-AILabs [168] is widely utilized by TTS programs, such as Deep-Voice 3 [127]. The M-AILABS dataset contains 999 h and 32 min of audio. Many native speakers of these nine languages contributed to the creation of this dataset.

AR-DAD, this is a fabricated audio file of Arabic speakers that was obtained from the audio site of the Holy Quran and was given the name Ar-DAD Arabic Diversified Audio [159]. It features both the authentic and imitated readings of the Quran, On the other hand, the audio speech includes 30 readers from Arabic countries and 12 imitators. More specifically, the reciters were individual males who were native speakers of Arabic and hail from the nations of the United Arab Emirates, Yemen, Egypt, Kuwait, Sudan, and Saudi Arabia. The data comprised 15,810 real samples and 379 fake samples, each of which was ten seconds long. Classical Arabic (CA) is a moniker given to the language of the dataset because it is written in Arabic.

H-Voice (Histograms Voice), Recent work has resulted in the creation of a dataset known as H-Voice [169], which uses synthesized and imitative voices to speak languages including English, French, Tagalog, Portuguese, and Spanish. In the PNG format, we find the samples that were originally stored in a histogram. There of 6672 samples and a plethora of subfolders were collected. However, the total number of samples consisted of both natural and artificially created variants (3,332 actual and 3,264 fake samples) and natural and synthetically created variants (four real and 72 fake samples). Deep Voice 3, which is utilized to generate synthetic-based files, is freely accessible to the public.

FakeAVCeleb: The FakeAVCeleb dataset [163] is an innovative and limited dataset of English speakers, created using the SV2TTS tool. A synthetic process was used to create the dataset. It includes 20,490 samples, 490 of which are authentic and the remaining 20,000 are fake. Samples were available in the MP3 format and last precisely seven seconds each.

ADD, The audio deep synthesis detection (ADD) competition [170] unveiled a novel dataset that aims to identify synthetic-based audio. The dataset comprises three distinct categories, namely low-quality fake audio detection (LF), partial fake audio detection (PF), and fake audio games (FG).

The LF dataset comprises a total of 1052 audio samples, predominantly of synthetic origin. On the other hand, the PF dataset encompasses 300 authentic vocal recordings, alongside 700 artificially generated words that are accompanied by ambient noise. The dataset is readily accessible to the public and has been curated in the Chinese language, rendering it available for utilization by researchers.

IV. CONCLUSION AND DISCUSSION

Numerous investigations have been carried out to identify audio deepfakes utilizing diverse (DL) methodologies. The aforementioned methodologies encompass (DNNs), (CNNs), and (CRNNs). The Human Log-Likelihoods (HLLs) methodology, which utilizes a (DNN) classifier, exhibited superior performance compared to the conventional GMM technique. Specifically, the HLLs approach achieved an equal error rate (EER) of 12.24% on the ASV spoof challenge 2015 dataset. Although the utilization of the ASSERT technique relying on a Squeeze-Excitation Network and a residual CNN-based approach exhibited encouraging outcomes, both methods encountered challenges with generalization. By way of comparison, the utilization of transfer learning and the ResNet-34 methodology within a framework yielded the most optimal outcomes, as evidenced by an EER of 5.32% and t-DCF of 0.11514% on the ASVspoof 2019 dataset.

The ASVspoof 2019 dataset was evaluated using the Temporal Convolutional Network and Spatial Transformer Network, resulting in accuracy rates of 92% and 80%, respectively. In addition, the study produced two CRNN-based models, wherein one model demonstrated superior performance compared to the other by 0.132% in the Tandem Decision Cost Function (t-DCF) and 4.27% in Equal Error Rate (EER) on the identical dataset. A technique for alignment, which utilized three classification models, was also suggested and exhibited satisfactory performance on the ASVspoof 2019 dataset.

The utilization of the transfer learning and ResNet-34 technique framework has demonstrated superior performance, as evidenced by its attainment of the lowest EER and t-DCF on the ASVspoof 2019 dataset.

REFERENCES

[1] Z. Khanjani, G. Watson, and V. P. Janeja, "How deep are the fakes? Focusing on audio deepfake: A survey," 2020, *arXiv:2111.14203*.

[2] X. Zhang and A. A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion," *Inf. Process. Manag.*, vol. 57, no. 2, Mar. 2020, Art. no. 102025, doi: 10.1016/j.ipm.2019.03.004.

[3] J. Shin, L. Jian, K. Driscoll, and F. Bar, "The diffusion of misinformation on social media: Temporal pattern, message, and source," *Comput. Hum. Behav.*, vol. 83, pp. 278–287, Jun. 2018, doi: 10.1016/j.chb.2018.02.008.

[4] G. Gravanis, A. Vakali, K. Diamantaras, and P. Karadais, "Behind the cues: A benchmarking study for fake news detection," *Exp. Syst. Appl.*, vol. 128, pp. 201–213, Aug. 2019, doi: 10.1016/j.eswa.2019.03.036.

[5] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," *Inf. Sci.*, vol. 497, pp. 38–55, Sep. 2019, doi: 10.1016/j.ins.2019.05.035.

[6] S. Lyu, "DeepFake detection: Current challenges and next steps," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, London, U.K., 2020, pp. 1–6, doi: 10.1109/ICMEW46912.2020.9105991.

[7] N. Diakopoulos and D. Johnson, "Anticipating and addressing the ethical implications of deepfakes in the context of elections," *New Media Soc.*, vol. 23, no. 7, pp. 2072–2098, Jul. 2021, doi: 10.1177/1461444820925811.

[8] Y. Rodríguez-Ortega, D. M. Ballesteros, and D. Renza, "A machine learning model to detect fake voice," in *Applied Informatics* (Communications in Computer and Information Science). Springer, 2020, pp. 3–13.

[9] T. Chen, A. Kumar, P. Nagarsheth, G. Sivaraman, and E. Khoury, "Generalization of audio deepfake detection," in *Proc. Speaker Lang. Recognit. Workshop (Odyssey)*, Tokyo, Japan, Nov. 2020, pp. 132–137, doi: 10.21437/odyssey.2020-19.

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

[11] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–13, Aug. 2017, doi: 10.1145/3072959.3073640.

[12] *Catherine Stupp Fraudsters Used AI to Mimic CEO's Voice in Unusual Cybercrime Case*. Accessed: Jul. 15, 2022. [Online]. Available: https://www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402

[13] Z. Almutairi and H. Elgibreen, "A review of modern audio deepfake detection methods: Challenges and future directions," *Algorithms*, vol. 15, no. 5, p. 155, May 2022, doi: 10.3390/a15050155.

[14] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The DeepFake detection challenge (DFDC) dataset," 2020, *arXiv:2006.07397*.

[15] B. Malolan, A. Parekh, and F. Kazi, "Explainable deep-fake detection using visual interpretability methods," in *Proc. 3rd Int. Conf. Inf. Comput. Technol. (ICICT)*, Mar. 2020, pp. 289–293, doi: 10.1109/ICICT50521.2020.00051.

[16] E. C. Tandoc, D. Lim, and R. Ling, "Diffusion of disinformation: How social media users respond to fake news and why," *Journalism*, vol. 21, no. 3, pp. 381–398, Mar. 2020, doi: 10.1177/1464884919868325.

[17] X. Zhou and R. Zafarani, "A survey of fake news: Fundamental theories, detection methods, and opportunities," *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–40, Sep. 2021, doi: 10.1145/3395046.

[18] C.-S. Atodiresei, A. Tanaselea, and A. Iftene, "Identifying fake news and fake users on Twitter," *Proc. Comput. Sci.*, vol. 126, pp. 451–461, Jan. 2018, doi: 10.1016/j.procs.2018.07.279.

[19] M. Aldwairi and A. Alwahedi, "Detecting fake news in social media networks," *Proc. Comput. Sci.*, vol. 141, pp. 215–222, Jan. 2018, doi: 10.1016/j.procs.2018.10.171.

[20] C. Sindermann, A. Cooper, and C. Montag, "A short review on susceptibility to falling for fake political news," *Current Opinion Psychol.*, vol. 36, pp. 44–48, Dec. 2020.

[21] D. D. Parsons. (2020). *The Impact of Fake News on Company Value: Evidence From Evidence From Tesla and Galena Biopharma*. [Online]. Available: https://trace.tennessee.edu/utk_chanhonoproj

[22] D. Ippolito, D. Duckworth, C. Callison-Burch, and D. Eck, "Automatic detection of generated text is easiest when humans are fooled," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 1808–1822, doi: 10.18653/v1/2020.acl-main.164.

[23] D. I. Adelani, H. Mai, F. Fang, H. H. Nguyen, J. Yamagishi, and I. Echizen, "Generating sentiment-preserving fake online reviews using neural language models and their human- and machine-based detection," in *Advanced Information Networking and Applications* (Advances in Intelligent Systems and Computing), vol. 1151. Springer, 2020, pp. 1341–1354, doi: 10.1007/978-3-030-44041-1_114.

[24] T. Fagni, F. Falchi, M. Gambini, A. Martella, and M. Tesconi, "TweepFake: About detecting deepfake tweets," *PLoS ONE*, vol. 16, no. 5, May 2021, Art. no. e0251415, doi: 10.1371/journal.pone.0251415.

[25] D. Dukic, D. Keca, and D. Stipic, "Are you human? Detecting bots on Twitter using BERT," in *Proc. IEEE 7th Int. Conf. Data Sci. Adv. Analytics (DSAA)*, Oct. 2020, pp. 631–636, doi: 10.1109/DSAA49011.2020.00089.

[26] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007, doi: 10.1109/TASL.2007.907344.

[27] L. Guarnera, O. Giudice, and S. Battiato. *DeepFake Detection by Analyzing Convolutional Traces*. Accessed: Dec. 17, 2022. [Online]. Available: https://www.Guarnera_DeepFake_Detection_by_Analyzing_Convolutional_Traces_CVPRW_2020_paper.pdf

[28] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. Efros, "Detecting photoshopped faces by scripting photoshop," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10071–10080, doi: 10.1109/ICCV.2019.01017.

[29] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proença, and J. Fierrez, "GANprintR: Improved fakes and evaluation of the state of the art in face manipulation detection," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 5, pp. 1038–1048, Aug. 2020, doi: 10.1109/JSTSP.2020.3007250.

[30] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face X-ray for more general face forgery detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5000–5009, doi: 10.1109/CVPR42600.2020.00505.

[31] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5780–5789, doi: 10.1109/CVPR42600.2020.00582.

[32] T. Khakhulin, V. Sklyarova, V. Lempitsky, and E. Zakharov, "Realistic one-shot mesh-based head avatars," 2022, *arXiv:2206.08343*.

[33] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt, "Real-time expression transfer for facial reenactment," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–14, Nov. 2015.

[34] K. R. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. V. Jawahar, "A lip sync expert is all you need for speech to lip generation in the wild,"in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 484–492, doi: 10.1145/3394171.3413532.

[35] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797, doi: 10.1109/CVPR.2018.00916.

[36] Y. Choi, Y. Uh, J. Yoo, and J. W. Ha, "StarGAN v2: Diverse image synthesis for multiple domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 8188–8197.

[37] R. Huang, S. Zhang, T. Li, and R. He, "Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2019, pp. 2439–2448.

[38] G. Y. Kang, Y. P. Feng, R. K. Wang, and Z. M. Lu, "Edge and feature points based video intra-frame passive-blind copy-paste forgery detection," *J. Netw. Intell.*, vol. 6, no. 3, pp. 637–645, 2021.

[39] G. Singh and K. Singh, "Chroma key foreground forgery detection under various attacks in digital video based on frame edge identification," *Multimedia Tools Appl.*, vol. 81, no. 1, pp. 1419–1446, Jan. 2022, doi: 10.1007/s11042-021-11380-3.

[40] O. Alamayreh and M. Barni, "Detection of GAN-synthesized street videos," in *Proc. 29th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2021, pp. 811–815, doi: 10.23919/EUSIPCO54536.2021.9616262.

[41] V. Kumar, A. Singh, V. Kansal, and M. Gaur, "A comprehensive survey on passive video forgery detection techniques," in *Recent Studies on Computational Intelligence* (Studies in Computational Intelligence), vol. 921. Singapore: Springer, 2021, pp. 39–57, doi: 10.1007/978-981-15-8469-5_4.

[42] C. Rathgeb, R. Tolosana, R. Vera-Rodriguez, and C. Busch, *Advances in Computer Vision and Pattern Recognition Handbook of Digital Face Manipulation and Detection From DeepFakes to Morphing Attacks*. Springer, 2022.

[43] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6, doi: 10.1109/AVSS.2018.8639163.

[44] D. Güera, "Media forensics using machine learning approaches," Doctoral dissertation, Dept. Elect. Comput. Eng., Purdue Univ., West Lafayette, IN, USA, 2019.

[45] D. M. Montserrat, H. Hao, S. K. Yarlagadda, S. Baireddy, R. Shao, J. Horváth, E. Bartusiak, J. Yang, D. Güera, F. Zhu, and E. J. Delp, "Deepfakes detection with automatic face weighting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 2851–2859.

[46] L. Bondi, E. Daniele Cannas, P. Bestagini, and S. Tubaro, "Training strategies and data augmentations in CNN-based DeepFake video detection," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2020, pp. 1–6, doi: 10.1109/WIFS49906.2020.9360901.

[47] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics networks for deepfake detection," in *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*. Berlin, Germany: Springer, 2022.

[48] N. M. Müller, K. Pizzi, and J. Williams, "Human perception of audio deepfakes," 2021, *arXiv:2107.09667*.

[49] B. Chesney and D. Citron, "Deep fakes: A looming challenge for privacy, democracy, and national security," *Calif. Law Rev.*, vol. 107, no. 6, pp. 1753–1820, 2019, doi: 10.15779/Z38RV0D15J.

[50] M. R. Al-Mousa, N. A. Sweerky, G. Samara, M. Alghanim, A. S. I. Hussein, and B. Qadoumi, "General countermeasures of anti-forensics categories," in *Proc. Global Congr. Electr. Eng. (GC-ElecEng)*, Dec. 2021, pp. 5–10, doi: 10.1109/GC-ElecEng52322.2021.9788230.

[51] G. C. Kessler, "Anti-forensics and the digital investigator," in *Proc. 5th Aust. Digit. Forensics Conf.*, 2007, pp. 1–7, doi: 10.4225/75/57ad39ee7ff25.

[52] T. Qazi, K. Hayat, S. U. Khan, S. A. Madani, I. A. Khan, J. Kolodziej, H. Li, W. Lin, K. C. Yow, and C. Xu, "Survey on blind image forgery detection," *IET Image Process.*, vol. 7, no. 7, pp. 660–670, Oct. 2013, doi: 10.1049/iet-ipr.2012.0388.

[53] K. Hayat and T. Qazi, "Forgery detection in digital images via discrete wavelet and discrete cosine transforms," *Comput. Electr. Eng.*, vol. 62, pp. 448–458, Aug. 2017, doi: 10.1016/j.compeleceng.2017.03.013.

[54] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," 2021, *arXiv:2106.06103*.

[55] K. Zhou, B. Sisman, R. Liu, and H. Li, "Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 920–924, doi: 10.1109/ICASSP39728.2021.9413391.

[56] G. Hua, A. B. J. Teoh, and H. Zhang, "Towards end-to-end synthetic speech detection," *IEEE Signal Process. Lett.*, vol. 28, pp. 1265–1269, 2021, doi: 10.1109/LSP.2021.3089437.

[57] A. Chintha, B. Thai, S. J. Sohrawardi, K. Bhatt, A. Hickerson, M. Wright, and R. Ptucha, "Recurrent convolutional structures for audio spoof and video deepfake detection," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 5, pp. 1024–1037, Aug. 2020, doi: 10.1109/JSTSP.2020.2999185.

[58] J. Volkmann, S. S. Stevens, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *J. Acoust. Soc. Amer.*, vol. 8, p. 208, Jan. 1937, doi: 10.1121/1.1901999.

[59] E. R. Bartusiak and E. J. Delp, "Frequency domain-based detection of generated audio," *Electron. Imag.*, vol. 33, no. 4, pp. 1–7, Jan. 2021.

[60] E. R. Bartusiak and E. J. Delp, "Synthesized speech detection using convolutional transformer-based spectrogram analysis," in *Proc. 55th Asilomar Conf. Signals, Syst., Comput.*, Oct. 2021, pp. 1426–1430, doi: 10.1109/IEEECONF53345.2021.9723142.

[61] J. Khochare, C. Joshi, B. Yenarkar, S. Suratkar, and F. Kazi, "A deep learning framework for audio deepfake detection," *Arabian J. Sci. Eng.*, vol. 47, no. 3, pp. 3447–3458, Mar. 2022, doi: 10.1007/s13369-021-06297-w.

[62] K. Bhagtani, A. K. S. Yadav, E. R. Bartusiak, Z. Xiang, R. Shao, S. Baireddy, and E. J. Delp, "An overview of recent work in multimedia forensics," in *Proc. IEEE 5th Int. Conf. Multimedia Inf. Process. Retr.*, Aug. 2022, pp. 324–329, doi: 10.1109/MIPR54900.2022.00064.

[63] F. Akdeniz and Y. Becerikli, "Detection of copy-move forgery in audio signal with Mel Frequency and Delta-Mel Frequency Kepstrum Coefficients," in *Proc. Innov. Intell. Syst. Appl. Conf. (ASYU)*, Oct. 2021, pp. 1–6, doi: 10.1109/ASYU52992.2021.9598977.

[64] M. Todisco, H. Delgado, and N. Evans, "Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Comput. Speech Lang.*, vol. 45, pp. 516–535, Sep. 2017, doi: 10.1016/j.csl.2017.01.001.

[65] Q. Yan, R. Yang, and J. Huang, "Robust copy-move detection of speech recording using similarities of pitch and formant," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 9, pp. 2331–2341, Sep. 2019, doi: 10.1109/TIFS.2019.2895965.

[66] F. Hassan and A. Javed, "Voice spoofing countermeasure for synthetic speech detection," in *Proc. Int. Conf. Artif. Intell. (ICAI)*, Apr. 2021, pp. 209–212, doi: 10.1109/ICAI52203.2021.9445238.

[67] R. K. Das, J. Yang, and H. Li, "Long range acoustic features for spoofed speech detection," in *Proc. INTERSPEECH*, 2019, pp. 1058–1062, doi: 10.21437/Interspeech.2019-1887.

[68] J. Li, H. Wang, P. He, S. M. Abdullahi, and B. Li, "Long-term variable Q transform: A novel time-frequency transform algorithm for synthetic speech detection," *Digit. Signal Process.*, vol. 120, Jan. 2022, Art. no. 103256, doi: 10.1016/j.dsp.2021.103256.

[69] X. Li, N. Li, C. Weng, X. Liu, D. Su, D. Yu, and H. Meng, "Replay and synthetic speech detection with Res2Net architecture," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 6354–6358, doi: 10.1109/ICASSP39728.2021.9413828.

[70] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 110, 2018, pp. 10019–10029.

[71] J. Lorenzo-Trueba, F. Fang, X. Wang, I. Echizen, J. Yamagishi, and T. Kinnunen, "Can we steal your vocal identity from the internet: Initial investigation of cloning Obama's voice using GAN, WaveNet and low-quality found data," in *Proc. Speaker Lang. Recognit. Workshop*, 2018, pp. 240–247, doi: 10.21437/odyssey.2018-34.

[72] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, and L. Juvela, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Comput. Speech Lang.*, vol. 64, Nov. 2020, Art. no. 101114, doi: 10.1016/j.csl.2020.101114.

[73] M. Swan. *WaveNet: A Generative Model for Raw Audio*. Accessed: Feb. 23, 2023. [Online]. Available: https://research.google/pubs/pub45774/

[74] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, Aug. 2017, pp. 4006–4010, doi: 10.21437/interspeech.2017-1452.

[75] Z. Jin, G. J. Mysore, S. Diverdi, J. Lu, and A. Finkelstein, "VoCo: Text-based insertion and replacement in audio narration," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–13, Aug. 2017, doi: 10.1145/3072959.3073702.

[76] *Audacity*. accessed: Sep. 9, 2020. [Online]. Available: https://www.audacityteam.org

[77] J. Damiani. *A Voice Deepfake Was Used To Scam A CEO Out Of $243,000*. Accessed: Sep. 6, 2020. [Online]. Available: https://www.forbes.com/sites/jessedamiani/2019/09/03/a-voice-deepfake-was-used-to-scam-a-ceo-out-of-243000/

[78] A. Leung. *NVIDIA Reveals That Part of Its CEO's Keynote Presentation Was Deepfaked*. Accessed: Aug. 29, 2021. [Online]. Available: https://hypebeast.com/2021/8/nvidia-deepfake-jensen-huang-omniverse-keynote-video

[79] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2Wav: End-to-end speech synthesis," in *Proc. 5th Int. Conf. Learn. Represent.*, 2015, pp. 1–6.

[80] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 132–157, 2021, doi: 10.1109/TASLP.2020.3038524.

[81] P. Partila, J. Tovarek, G. H. Ilk, J. Rozhon, and M. Voznak, "Deep learning serves voice cloning: How vulnerable are automatic speaker vulnerable systems to Spoofing trials?" *IEEE Commun. Mag.*, vol. 58, no. 2, pp. 100–105, Feb. 2020, doi: 10.1109/MCOM.001.1900396.

[82] R. Natsume, T. Yatagawa, and S. Morishima, "RSGAN: Face swapping and editing using face and hair representation in latent spaces," 2009, *arXiv:1804.03447*.

[83] *Faceswap-GAN*. Accessed: May 29, 2022. [Online]. Available: https://github.com/shaoanlu/faceswap-GAN

[84] R. Natsume, T. Yatagawa, and S. Morishima, "RSGAN: Face swapping and editing using face and hair representation in latent spaces," in *Proc. ACM SIGGRAPH Posters*, 2018, pp. 1–2, Art. no. 69, doi: 10.1145/3230744.3230818.

[85] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," 2016, *arXiv:1609.03499*.

[86] Y. Taigman, L. Wolf, A. Polyak, and E. Nachmani, "VoiceLoop: Voice fitting and synthesis via a phonological loop," in *Proc. 6th Int. Conf. Learn. Represent.*, 2018, pp. 1–14.

[87] I. Korshunova, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2017, pp. 3677–3685.

[88] Y. Nirkin, Y. Keller, and T. Hassner, "FSGAN: Subject agnostic face swapping and reenactment," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7183–7192, doi: 10.1109/ICCV.2019.00728.

[89] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 4480–4490.

[90] J. Cong, S. Yang, L. Xie, G. Yu, and G. Wan, "Data efficient voice cloning from noisy samples with domain adversarial training," in *Proc. Interspeech*, Oct. 2020, pp. 811–815, doi: 10.21437/interspeech.2020-2530.

[91] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *Proc. 6th Int. Conf. Learn. Represent.*, 2018, pp. 1–16.

[92] A. Van Den, N. Kalchbrenner, L. Espeholt, O. Vinyals, and A. Graves, "Conditional image generation with PixelCNN decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4790–4798. [Online]. Available: https://papers.nips.cc/paper/6527-conditional-image-generation-with-pixelcnn-decoders.pdf

[93] A. Van Den Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. Driessche, E. Lockhart, L. Cobo, F. Stimberg, and N. Casagrande, "Parallel WaveNet: Fast high-fidelity speech synthesis," in *Proc. 35th Int. Conf. Mach. Learn. (ICML)*, vol. 9, 2018, pp. 6270–6278.

[94] S. O. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman, and S. Sengupta, "Deep voice: Real-time neural text-to-speech," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 2963–2971.

[95] S. O. Arik and J. Miller, "Deep voice 2: Multi-speaker neural text-to-speech," in *Proc. NIPS*, vol. 1, 2017, pp. 1–9.

[96] Y. Yasuda, X. Wang, S. Takaki, and J. Yamagishi, "Investigation of enhanced tacotron text-to-speech synthesis systems with self-attention for pitch accent language," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6905–6909, doi: 10.1109/ICASSP.2019.8682353.

[97] Y. Lee, T. Kim, and S.-Y. Lee, "Voice imitating text-to-speech neural networks," 2018, *arXiv:1806.00927*.

[98] H.-T. Luong and J. Yamagishi, "NAUTILUS: A versatile voice cloning system," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 2967–2981, 2020, doi: 10.1109/TASLP.2020.3034994.

[99] Y. Chen, Y. Assael, B. Shillingford, D. Budden, S. Reed, H. Zen, Q. Wang, L. C. Cobo, A. Trask, B. Laurie, and C. Gulcehre, "Sample efficient adaptive text-to-speech," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–16.

[100] Z. Yi, W. C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice conversion challenge 2020—Intra-lingual semi-parallel and cross-lingual voice conversion," in *Proc. Joint Workshop Blizzard Challenge Voice Convers. Challenge*, 2020, pp. 80–98, doi: 10.21437/vcc_bc.2020-14.

[101] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," in *Proc. Speaker Lang. Recognit. Workshop*, 2018, pp. 195–202, doi: 10.21437/odyssey.2018-28.

[102] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 131–142, Mar. 1998, doi: 10.1109/89.661472.

[103] E. Helander, H. Silen, T. Virtanen, and M. Gabbouj, "Voice conversion using dynamic kernel partial least squares regression," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 3, pp. 806–817, Mar. 2012, doi: 10.1109/TASL.2011.2165944.

[104] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 10, pp. 1506–1521, Oct. 2014, doi: 10.1109/TASLP.2014.2333242.

[105] T. Nakashika, T. Takiguchi, and Y. Ariki, "High-order sequence modeling using speaker-dependent recurrent temporal restricted Boltzmann machines for voice conversion," in *Proc. Interspeech*, Sep. 2014, pp. 2278–2282, doi: 10.21437/interspeech.2014-447.

[106] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4869–4873, doi: 10.1109/ICASSP.2015.7178896.

[107] H. Ming, D. Huang, L. Xie, J. Wu, M. Dong, and H. Li, "Deep bidirectional LSTM modeling of timbre and prosody for emotional voice conversion," in *Proc. Interspeech*, Sep. 2016, pp. 2453–2457, doi: 10.21437/interspeech.2016-1053.

[108] J. Wu, Z. Wu, and L. Xie, "On the use of I-vectors and average voice model for voice conversion without parallel data," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Dec. 2016, pp. 1–6, doi: 10.1109/APSIPA.2016.7820901.

[109] L.-J. Liu, Z.-H. Ling, Y. Jiang, M. Zhou, and L.-R. Dai, "WaveNet vocoder with limited training data for voice conversion," in *Proc. Interspeech*, Sep. 2018, pp. 1983–1987, doi: 10.21437/interspeech.2018-1190.

[110] P.-C. Hsu, C.-H. Wang, A. T. Liu, and H.-Y. Lee, "Towards robust neural vocoding for speech generation: A survey," 2019, *arXiv:1912.02461*.

[111] T. Kaneko and H. Kameoka, "CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 2100–2104, doi: 10.23919/EUSIPCO.2018.8553236.

[112] M. Zhang, B. Sisman, L. Zhao, and H. Li, "DeepConversion: Voice conversion with limited parallel training data," *Speech Commun.*, vol. 122, pp. 31–43, Sep. 2020, doi: 10.1016/j.specom.2020.05.004.

[113] W.-C. Huang, H. Luo, H.-T. Hwang, C.-C. Lo, Y.-H. Peng, Y. Tsao, and H.-M. Wang, "Unsupervised representation disentanglement using cross domain features and adversarial learning in variational autoencoder based voice conversion," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 4, no. 4, pp. 468–479, Aug. 2020, doi: 10.1109/TETCI.2020.2977678.

[114] J. Chorowski, R. J. Weiss, S. Bengio, and A. Van Den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 12, pp. 2041–2053, Dec. 2019.

[115] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding Wasserstein generative adversarial networks," in *Proc. Interspeech*, Aug. 2017, pp. 3364–3368, doi: 10.21437/interspeech.2017-63.

[116] F. Fang, J. Yamagishi, I. Echizen, and J. Lorenzo-Trueba, "High-quality nonparallel voice conversion based on cycle-consistent adversarial network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 5279–5283, doi: 10.1109/ICASSP.2018.8462342.

[117] K. Tanaka, T. Kaneko, N. Hojo, and H. Kameoka, "Synthetic-to-natural speech waveform conversion using cycle-consistent adversarial networks," in *Proc. IEEE Spoken Lang. Technol. Workshop*, Dec. 2018, pp. 632–639.

[118] J.-C. Chou, C.-C. Yeh, H.-Y. Lee, and L.-S. Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," in *Proc. Interspeech*, Sep. 2018, pp. 501–505, doi: 10.21437/interspeech.2018-1830.

[119] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6199–6203, doi: 10.1109/ICASSP40776.2020.9053795.

[120] K. Tanaka, H. Kameoka, T. Kaneko, and N. Hojo, "AttS2S-VC: Sequence-to-sequence voice conversion with attention and context preservation mechanisms, NTT Corporation, Japan," in *Proc. IEEE Int. Conf. Acoustics, Speech Signal Process.*, May 2019, pp. 6805–6809.

[121] S.-W. Park, D.-Y. Kim, and M.-C. Joe, "Cotatron: Transcription-guided speech encoder for Any-to-Many voice conversion without parallel data," in *Proc. Interspeech*, Oct. 2020, pp. 4696–4700, doi: 10.21437/interspeech.2020-1542.

[122] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice transformer network: Sequence-to-Sequence voice conversion using transformer with text-to-speech pretraining," in *Proc. Interspeech*, Oct. 2020, pp. 4676–4680, doi: 10.21437/interspeech.2020-1066.

[123] H. Lu, Z. Wu, D. Dai, R. Li, S. Kang, J. Jia, and H. Meng, "One-shot voice conversion with global speaker embeddings," in *Proc. Interspeech*, 2021, pp. 669–673.

[124] R. Gontijo Lopes, S. Fenu, and T. Starner, "Data-free knowledge distillation for deep neural networks," 2017, *arXiv:1710.07535*.

[125] T. H. Huang, J. H. Lin, and H. Y. Lee, "How far are we from robust voice conversion: A survey," in *Proc. IEEE Spoken Lang. Technol. Workshop*, Jan. 2021, pp. 514–521, doi: 10.1109/SLT48900.2021.9383498.

[126] N. Li, D. Tuo, D. Su, Z. Li, and D. Yu, "Deep discriminative embeddings for duration robust speaker verification," in *Proc. Interspeech*, Sep. 2018, pp. 2262–2266.

[127] J.-C. Chou and H.-Y. Lee, "One-shot voice conversion by separating speaker and content representations with instance normalization," in *Proc. Interspeech*, Sep. 2019, pp. 664–668, doi: 10.21437/interspeech.2019-2663.

[128] Y. Rebryk and S. Beliaev, "ConVoice: Real-time zero-shot voice style transfer with convolutional network," 2020, *arXiv:2005.07815*.

[129] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," 2016, pp. 5206–5210.

[130] M. R. Kamble, H. B. Sailor, H. A. Patil, and H. Li, "Advances in anti-spoofing: From the perspective of ASVspoof challenges," *APSIPA Trans. Signal Inf. Process.*, vol. 9, p. e2, Jan. 2020, doi: 10.1017/ATSIP.2019.21.

[131] J. Yi, Y. Bai, J. Tao, H. Ma, Z. Tian, C. Wang, T. Wang, and R. Fu, "Half-truth: A partially fake audio detection dataset," in *Proc. Interspeech*, Aug. 2021, pp. 2683–2687, doi: 10.21437/interspeech.2021-930.

[132] P. R Aravind, U. Nechiyil, and N. Paramparambath, "Audio spoofing verification using deep convolutional neural networks by transfer learning," 2020, *arXiv:2008.03464*.

[133] J. Monteiro, J. Alam, and T. H. Falk, "Generalized end-to-end detection of spoofing attacks to automatic speaker recognizers," *Comput. Speech Lang.*, vol. 63, Sep. 2020, Art. no. 101096, doi: 10.1016/j.csl.2020.101096.

[134] Y. Gao, T. Vuong, M. Elyasi, G. Bharaj, and R. Singh, "Generalized spoofing detection inspired from audio generation artifacts," in *Proc. Interspeech*, Aug. 2021, pp. 3691–3695, doi: 10.21437/interspeech.2021-1705.

[135] Z. Zhang, X. Yi, and X. Zhao, "Fake speech detection using residual network with transformer encoder," in *Proc. ACM Workshop Inf. Hiding Multimedia Secur.*, Jun. 2021, pp. 13–22, doi: 10.1145/3437880.3460408.

[136] R. K. Das, J. Yang, and H. Li, "Data augmentation with signal companding for detection of logical access attacks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Jun. 2021, pp. 6349–6353.

[137] M. Aljasem, A. Irtaza, H. Malik, N. Saba, A. Javed, K. M. Malik, and M. Meharmohammadi, "Secure automatic speaker verification (SASV) system through sm-ALTP features and asymmetric bagging," *IEEE Trans. Inf. Forensics Security.*, vol. 16, pp. 3524–3537, 2021, doi: 10.1109/TIFS.2021.3082303.

[138] H. Ma, J. Yi, J. Tao, Y. Bai, Z. Tian, and C. Wang, "Continual learning for fake audio detection," in *Proc. Interspeech*, Aug. 2021, pp. 1748–1752, doi: 10.21437/interspeech.2021-794.

[139] C. Borrelli, P. Bestagini, F. Antonacci, A. Sarti, and S. Tubaro, "Synthetic speech detection through short-term and long-term prediction traces," *EURASIP J. Inf. Secur.*, vol. 2021, no. 1, pp. 1–14, Dec. 2021, doi: 10.1186/s13635-021-00116-3.

[140] E. A. AlBadawy, S. Lyu, and H. Farid, "Detecting AI-synthesized speech using bispectral analysis," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, Jun. 2019, pp. 104–109.

[141] A. K. Singh and P. Singh, "Detection of AI-synthesized speech using cepstral & bispectral statistics," in *Proc. IEEE 4th Int. Conf. Multimedia Inf. Process. Retr. (MIPR)*, Sep. 2021, pp. 412–417.

[142] H. Malik and R. Changalvala, "Fighting AI with AI: Fake speech detection using deep learning," in *Proc. AES Int. Conf.*, Jun. 2019, pp. 1–9.

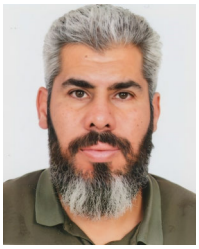
[143] L. Huang and C.-M. Pun, "Audio replay spoof attack detection by joint segment-based linear filter bank feature extraction and attention-enhanced DenseNet-BiLSTM network," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1813–1825, 2020, doi: 10.1109/TASLP.2020.2998870.

[144] Z. Wu, R. K. Das, J. Yang, and H. Li, "Light convolutional neural network with feature genuinization for detection of synthetic speech attacks," in *Proc. Interspeech*, Oct. 2020, pp. 1101–1105, doi: 10.21437/interspeech.2020-1810.

[145] Y. Zhang, F. Jiang, and Z. Duan, "One-class learning towards synthetic voice spoofing detection," *IEEE Signal Process. Lett.*, vol. 28, pp. 937–941, 2021, doi: 10.1109/LSP.2021.3076358.

[146] R. Wang, F. Juefei-Xu, Y. Huang, Q. Guo, X. Xie, L. Ma, and Y. Liu, "DeepSonar: Towards effective and robust detection of AI-synthesized fake voices," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 1207–1216, doi: 10.1145/3394171.3413716.

[147] R. Reimao and V. Tzerpos, "FoR: A dataset for synthetic speech detection," in *Proc. Int. Conf. Speech Technol. Human-Computer Dialogue (SpeD)*, Oct. 2019, pp. 1–10, doi: 10.1109/SPED.2019.8906599.
[148] H. Yu, Z.-H. Tan, Z. Ma, R. Martin, and J. Guo, "Spoofing detection in automatic speaker verification systems using DNN classifiers and dynamic acoustic features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4633–4644, Oct. 2018, doi: 10.1109/TNNLS.2017.2771947.
[149] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "A light convolutional GRU-RNN deep feature extractor for ASV spoofing detection," in *Proc. Interspeech*. 2021, pp. 1068–1072.
[150] C. I. Lai, N. Chen, J. Villalba, and N. Dehak, "ASSERT: Anti-spoofing with squeeze-excitation and residual networks," in *Proc. Interspeech*, Sep. 2019, pp. 1013–1017, doi: 10.21437/Interspeech.2019-1794.
[151] M. Alzantot, Z. Wang, and M. B. Srivastava, "Deep residual neural networks for audio spoofing detection," in *Proc. Interspeech*, Sep. 2019, pp. 1078–1082, doi: 10.21437/interspeech.2019-3174.
[152] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, and K. A. Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," in *Proc. Interspeech*, Sep. 2019, pp. 1008–1012, doi: 10.21437/interspeech.2019-2249.
[153] M. Shan and T. Tsai, "A cross-verification approach for protecting world leaders from fake and tampered audio," 2020, *arXiv:2010.12173*.
[154] R. L. P. C. Wijethunga, D. M. K. Matheesha, A. A. Noman, K. H. A. De Silva, M. Tissera, and L. Rupasinghe, "Deepfake audio detection: A deep learning based solution for group conversations," in *Proc. 2nd Int. Conf. Advancements Comput. (ICAC)*, vol. 1, Dec. 2020, pp. 192–197, doi: 10.1109/ICAC51239.2020.9357161.
[155] Z. Jiang, H. Zhu, L. Peng, W. Ding, and Y. Ren, "Self-supervised spoofing audio detection scheme," in *Proc. Interspeech*, 2020, pp. 4223–4227.
[156] N. Subramani and D. Rao, "Learning efficient representations for fake speech detection," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020, pp. 5859–5866, doi: 10.1609/aaai.v34i04.6044.
[157] Z. Lei, Y. Yang, C. Liu, and J. Ye, "Siamese convolutional neural network using Gaussian probability feature for spoofing speech detection," in *Proc. Interspeech*, Oct. 2020, pp. 1116–1120, doi: 10.21437/interspeech.2020-2723.
[158] M. Lataifeh, A. Elnagar, I. Shahin, and A. B. Nassif, "Arabic audio clips: Identification and discrimination of authentic cantillations from imitations," *Neurocomputing*, vol. 418, pp. 162–177, Dec. 2020, doi: 10.1016/j.neucom.2020.07.099.
[159] M. Lataifeh and A. Elnagar, "Ar-DAD: Arabic diversified audio dataset," *Data Brief*, vol. 33, Dec. 2020, Art. no. 106503, doi: 10.1016/j.dib.2020.106503.
[160] E. R. Bartusiak and E. J. Delp, "Frequency domain-based detection of generated audio," in *Proc. Electron. Imag., Soc. Imag. Sci. Technol.*, 2022, pp. 273–281.
[161] H. E. Delgado. *ASVspoof 2021*. Accessed: Aug. 6, 2021. [Online]. Available: https://www.asvspoof.org/
[162] T. Arif, A. Javed, M. Alhameed, F. Jeribi, and A. Tahir, "Voice spoofing countermeasure for logical access attacks detection," *IEEE Access*, vol. 9, pp. 162857–162868, 2021, doi: 10.1109/ACCESS.2021.3133134.
[163] H. Khalid, S. Tariq, M. Kim, and S. S. Woo, "FakeAVCeleb: A novel audio-video multimodal deepfake dataset," 2021, *arXiv:2108.05080*.
[164] H. Khalid, M. Kim, S. Tariq, and S. S. Woo, "Evaluation of an audio-video multimodal deepfake dataset using unimodal and multimodal detectors," in *Proc. 1st Workshop Synth. Multimedia-Audiovisual Deepfake Gener. Detection*, Oct. 2021, pp. 7–15.
[165] S. Camacho, D. M. Ballesteros, and D. Renza, "Fake speech recognition using deep learning," in *Applied Computer Sciences in Engineering* (Communications in Computer and Information Science), vol. 1431. Cham, Switzerland: Springer, 2021, pp. 38–48.
[166] Z. Almutairi and H. Elgibreen, "Detecting fake audio of Arabic speakers using self-supervised deep learning," *IEEE Access*, vol. 11, pp. 72134–72147, 2023, doi: 10.1109/ACCESS.2023.3286864.
[167] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proc. 12th Int. Conf. Lang. Resour. Eval. Conf.*, 2020, pp. 4218–4222.
[168] *The M-AILABS Speech Dataset*. Accessed: Feb. 25, 2021. [Online]. Available: https://www.caito.de/2019/01/the-m-ailabs-speech-dataset
[169] D. M. Ballesteros, Y. Rodriguez, and D. Renza, "A dataset of histograms of original and fake voice recordings (H-Voice)," *Data Brief*, vol. 29, Apr. 2020, Art. no. 105331, doi: 10.1016/j.dib.2020.105331.
[170] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang, Z. Tian, Y. Bai, C. Fan, S. Liang, S. Wang, S. Zhang, X. Yan, L. Xu, Z. Wen, and H. Li, "ADD 2022: The first audio deep synthesis detection challenge," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 9216–9220, doi: 10.1109/ICASSP43922.2022.9746939.



OUSAMA A. SHAABAN received the B.S. degree in computer maintenance from the College of Computer Technology, Tripoli, Libya, in 2003, and the M.S. degree in information technology from Universiti Utara Malaysia (UUM), Kedah, Malaysia, in 2015. He is currently pursuing the Ph.D. degree in computer engineering with Ankara Yıldırım Beyazıt University, Ankara, Turkey. His research interests include augmented reality, machine learning, and DL.

REMZI YILDIRIM received the B.S. and M.S. degrees in electronics and computer engineering from Gazi University, Ankara, Turkey, in 1988 and 1993, respectively, and the Ph.D. degree in electronics from Erciyes University, Kayseri, Turkey, in 1996. From 1999 to 2002, he was a Visiting Scholar with the Massachusetts Institute of Technology, Boston, MA, USA. From 2004 to 2005, he was with Liverpool University, U.K. He is currently a Professor with the Department of Computer Engineering, Ankara Yıldırım Beyazıt University, Ankara. He has published more than 100 journal articles and conference papers and 21 books. His research interests include optoelectronics, cyber-physical systems, and model checking.

ABUBAKER A. ALGUTTAR received the B.Sc. degree in computer science from Benghazi University, in 1993, and the M.Eng.Sc. degree in computer science and engineering and the M.T.M. degree in technology management from the University of New South Wales, Sydney, NSW, Australia, in 2004 and 2005, respectively. He is currently pursuing the Ph.D. degree in computer engineering with Ankara Yıldırım Beyazıt University, Ankara, Turkey. From 2006 to 2018, he was a Lecturer with the College of Information Technology, Misurata University, Misurata, Libya. His research interests include machine and DL, and fake news detection.