Deepfakes Audio Detection Leveraging Audio Spectrogram and Convolutional Neural Networks

Taiba Majid Wani(✉) and Irene Amerini

Sapienza University of Rome, Rome, Italy
{majid,amerini}@diag.uniroma1.it

Abstract. The proliferation of algorithms and commercial tools for the creation of synthetic audio has resulted in a significant increase in the amount of inaccurate information, particularly on social media platforms. As a direct result of this, efforts have been concentrated in recent years on identifying the presence of content of this kind. Despite this, there is still a long way to go until this problem is adequately addressed because of the growing naturalness of fake or synthetic audios. In this study, we proposed different networks configurations: a Custom Convolution Neural Network (cCNN) and two pretrained models (VGG16 and MobileNet) as well as end-to-end models to classify real and fake audios. An extensive experimental analysis was carried out on three classes of audio manipulation of the dataset FoR deepfake audio dataset. Also, we combined such sub-datasets to formulate a combined dataset FoR-combined to enhance the performance of the models. The experimental analysis shows that the proposed cCNN outperforms all the baseline models and other reference works with the highest accuracy of 97.23% on FoR-combined and sets new benchmarks for the datasets.

Keywords: Audio Deepfakes · FoR dataset · CNN · VGG16 · MobileNet

1 Introduction

Nowadays, we are said to be living in the "post-truth" era, which refers to a time where malicious actors can sway public opinion through disinformation in society. Disinformation is an active measure that can cause great damage, such as the manipulation of elections, the creation of conditions that could lead to war, the slandering of any individual, and so on. Recently, substantial advancements have been made in the development of deepfakes. This technology has the potential to be utilized in the dissemination of false information and may soon provide a significant risk in the form of fake news. Deepfakes are videos and audio that have been synthesized and created by Artificial Intelligence. Deepfakes are having an increasingly negative impact on people's ability to maintain their privacy and social security, as well as their authenticity [1]. Recent research has been centered on the identification of deepfake video, which has resulted in an adequate detection accuracy [2].

G. L. Foresti et al. (Eds.): ICIAP 2023, LNCS 14234, pp. 156–167, 2023.
https://doi.org/10.1007/978-3-031-43153-1_14

The identification of audio deepfake has received significantly less attention than the detection of video deepfakes. Utilizing deep learning algorithms, audio deepfakes focus on the production, editing, or synthesis of the target speaker's voice. The goal of such manipulations is to depict the speaker as saying something they have not actually spoken. Over the course of the past few years, voice manipulation has also developed into a very advanced art form [3]. Not only does creating synthetic voices present a threat to automated speaker verification systems, but they also present a threat to voice-controlled devices that have been developed for use in Internet of Things (IoT) contexts. Text-to-speech synthesis (also known as TTS) and voice conversion (VC) are two methods that can be used to generate fake voices [4]. A technology known as text-to-speech (TTS) synthesis may recreate the authentic-sounding voice of any speaker by modeling it after a text that is provided. Voice Conversion (VC) is the process of transforming the audio waveform of a source speaker into one that more closely resembles the speech of a target speaker. Voice synthesis using TTS and VC both produce computer-generated voices that are totally synthetic but are almost unrecognizable from real human speech. In [5] is presented a possible threat to biometric voice devices since the most recent speech synthesis algorithms can produce voices that have a high degree of similarity to a particular speaker. There is a significant risk that voice cloning may undermine public trust and provide criminals with the ability to influence corporate interactions or conversations that are private over the phone. It is anticipated that the incorporation of voice cloning into deepfakes will present a fresh difficulty for the identification of deepfakes [6]. Therefore, it is essential that, in contrast to the trend methodologies, which principally concentrate on identifying visual signal alterations, audio forgeries should also be investigated.

The ASVspoof datasets [7] are being utilized extensively in most of the research that is currently going on audio deepfake detection. However, using these datasets has several drawbacks, the most significant of which is that they do not contain any audio that was generated by the most recent text-to-speech algorithms. These algorithms produce audio that sounds more like human speech and may be unrecognizable to the ears of humans. It is possible that a more difficult issue will arise when attempting to differentiate such audio from true human-generated audio, which calls for the creation of reliable solutions. For this study, we made use of the FoR deepfake dataset [8] as it contains examples of audio generated by the most recent text-to-speech algorithms as well as original utterances. This dataset is the largest publicly available dataset and is selected in this study because it provides the true labels indicating whether the audio file is real or fake. These labels are used during model training and help the model to learn the patterns and features of deepfake audio.

The identification of audio deepfakes has made use of several different machine learning (ML) techniques in the literature. ML-based methods to detect deepfakes follow the conventional pipeline such as feature generation, extraction, and then classification. While approaches that are based on deep learning (DL) need less effort from humans to be put into feature engineering and have obtained very accurate results in detecting audio deepfakes, traditional methods still have their uses. In recent years, the DL approach based on Convolutional Neural Networks (CNNs) has been shown to exhibit remarkable performance in image processing benchmarking competitions, and computer vision,

because of its powerful learning capabilities [9]. CNNs are strong in their capacity to grasp spatiotemporal correlations and automatically learn data representations by utilizing numerous feature extraction phases.

For this reason, to leverage the CNN powerfulness, we have designed a Custom Convolution Neural Network (cCNN) detection model composed of four convolutional layers and two fully connected layers to prevent overfitting. We demonstrate the effectiveness of the proposed methodology through a comprehensive experimental evaluation over FoR deepfake dataset. We implemented pre-trained transfer learning models, VGG16 and MobileNet and in addition, trained these models end-to-end for extensive evaluation. The presented models take as input the mel spectrogram (a spectrogram where the frequencies are converted to the mel scale) generated from the three sub-datasets, for-norm, for-2-s, and for-rerecording, constituting the FoR dataset. Mel spectrograms are generated to capture the frequency content of an audio signal and the models can concentrate on the most crucial elements of the audio signal for the classification task. We also combined these sub-datasets into one dataset and named it FoR-combined. Data argumentation is performed on FoR-combined and use it for evaluation process.

2 Related Works

Recent developments in TTS and VC techniques have made audio deepfakes an increasingly dangerous threat to voice biometric interfaces and society. There are a few strategies within the realm of audio forensics to recognize them, but the existing studies are not completely efficient. In this section, we have reviewed recent works that have leveraged FoR dataset employing different ML and DL algorithms.

A novel DL architecture namely, DeepSonar based on layer-by-layer neuron behavior was proposed by Wang et. al. [10]. The model used binary classification for detection of fake and real speeches. A total of three datasets were used, FoR dataset in its original form and two datasets created by authors, Sprocket-VC (in English), made by using open-sourced tool sprocket and MC-TTS (in Chinese) using ancient Chinese poetry. The proposed model achieved an accuracy of 98.1% and EER of 2%.

Camacho et. al., presented a two-stage model for the recognition of fake speech [11]. The first stage included the transformation of raw data to scatter plots and modelling of data using CNN was carried out in other stages. CNN was trained on for-original version of FoR dataset and achieved an accuracy of 88% with 11% of EER. Kochare et.al., [12] implemented several machine learning techniques and two deep learning techniques, a temporal convolutional network (TCN) and a spatial transformer network (STN) for the detection of audio deepfakes using for-original dataset only. TCN and STN achieved an accuracy of 92% and 80% respectively.

The architecture of the proposed cCNN model is simpler and easier to interpret as compared to the state-of-art models [11] and [12], with each layer having a specific function. The balance of convolutional layers and the dropout helps to mitigate the risk of overfitting and hence the robustness of model.

Iqbal et. al., [13] proposed an approach based on selecting the best machine learning algorithm and optimal feature engineering. The feature preprocessing involved feature normalization and feature selection. The three sub-datasets of FoR, for-norm for-rerec

and for-2-s were used for training six machine learning techniques. eXtreme Gradient Boosting (XGB) obtained the highest average accuracy of 93%. Hamza et. al., [14] carried out detailed experiments on FoR dataset. Different ML algorithms using mel-frequency cepstral coefficients (MFCC) features were trained on 3 sub datasets of FoR dataset.

In this work, we designed a Custom Convolutional Neural Network (cCNN) using mel spectrograms as input features. The novelty of the presented architecture compared to the state-of-art works lies in the specific configuration of the layers and their hyperparameters. Mel spectrograms provide a visual representation of the intensity of different frequencies, allowing cCNNs to recognize various patterns and features to classify real and fake audio. Mel spectrograms also speed up training time and reduce overfitting, improving the models' overall performance. Further, the study has given the possibility of leveraging pre-trained models that were learnt from the well-known dataset ImageNet. These models are then transferred to the specific task of detecting real and fake audios though the use of the FoR dataset.

3 Methodology

The proposed methodology for the detection of real and fake audios is depicted in Fig. 1. Initially, the audio files present in the datasets are pre-processed and undergo a series of transformations. The processed audio files are converted into mel spectrograms facilitating an image-based approach for the required classification process. Finally, the generated mel spectrograms are given as input to the presented models such as the cCNN, VGG16 and MobileNet to detect the real and fake audios. VGG16 and MobileNet are used as pre-trained transfer learning networks as well as models trained end-to-end.

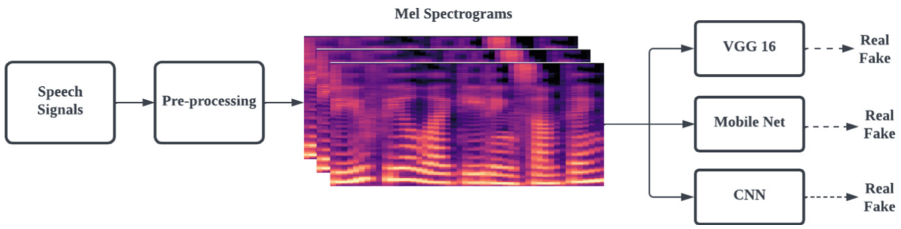


Fig. 1. Proposed methodology for the detection of real and fake audios.

3.1 Pre-processing

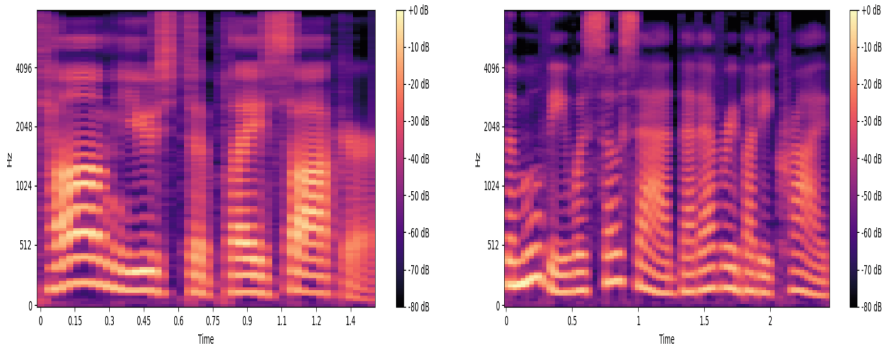
Preprocessing is a crucial step in the classification problems, as it converts unstructured data into structured format. FoR dataset consists of duplicate and 0-bit files, and these files affect the model's training and hence performance. We preprocessed the dataset and removed these files. The remaining files are converted into mel spectrograms.

Mel spectrogram applies a frequency-domain filter bank to audio signal that are windowed in time. Mel spectrogram employs the mel scale to simulate the non-linear

frequency perception of the human auditory system, which is logarithmic and collects the most perceptually significant information present in audios. Mel spectrogram is generated by dividing the audio signals into overlapping frames using the windowing function [15]. Short-term frequency transform (STFT) is calculated for obtaining the spectrograms, followed by using mel-scale filter banks to convert frequency axis to mel scale, as given in Eq. 1, where m is the mel scale and f is the frequency in Hertz. Lastly, the logarithm of filter bank energies is calculated to acquire the required melspectrogram.

$$\mathrm{m} = 2595 * \log 10(1 + \mathrm{f}/700) - 1 \tag{1}$$

In this study, we generated mel spectrograms of size $224 \times 224 \times 3$, using the Hanning window with size of 2048 and hop length of 512. The number of mel filter banks used was 224. In addition, logarithmic scaling and normalization were used to make the mel spectrograms more resistant to aberrations and background noise. Figure 2a and 2b represent the mel spectrograms of real and fake audio samples taken from the dataset. Mel spectrograms, with time on the x-axis and frequency on y-axis in Hertz (Hz), are expressed in decibels (dB) as they represent the logarithmic scale of the power of a signal.



(a) (b)

Fig. 2. Mel spectrograms of (a) real audio and (b) fake audio.

It can be noticed in Fig. 2a and 2b, that the mel spectrogram of real audio exhibits a consistent and richer spectral content with well-defined format pattern across different frequency bands while the mel spectrogram of fake audio has inconsistencies and unnatural spectral peaks and formats. During training, the network should learn to recognize these variations and efficiently classify audio as real or fake.

3.2 Custom Convolutional Neural Network (cCNN)

In computer vision, Convolutional Neural Networks are the most widely used deep learning technique because of their scalability and stability. In this work, we have proposed a Custom Convolutional Neural Network (cCNN). cCNNs specific layer structure, which

includes small filter sizes and max pooling, as well as the addition of a dropout layer to prevent overfitting, is the basis of its efficacy for distinguishing between real and fake audio. The mel spectrograms generated from the three sub-datasets of FoR dataset are taken as input and fed to the first layer of cCNN. The first convolution layer has a kernel size of (3 × 3) with 64 filter, second and third layer have 128 filters of kernel size (5 × 5) and last convolution layer has 256 filters with kernel size of (5 × 5). Each convolution layer is followed by a ReLU activation unit and pooling layer of size (2 × 2) with same padding and stride of 2. The dimensionality of the feature maps is significantly reduced by using smaller filter sizes of (3 × 3) and (5 × 5) and max pooling layers of (2x2) as compared to bigger filter sizes and pooling layers. Batch normalization is carried out after every convolution layer to increase the training speed and hence stability. The output of the last pooling layer is fed to the flatten layer in which the 3D volume is converted to a 1D vector. The flatten layer is followed by two fully connected layers with 512 and 1024 neurons respectively. A dropout layer is added after the first fully connected layer with a dropout ratio of 25% to avoid overfitting. This improves the model's ability to generalize to new data and hence performance. The last fully connected layer consists of SoftMax activation function performing the task of classification of real and fake audios (Fig. 3).

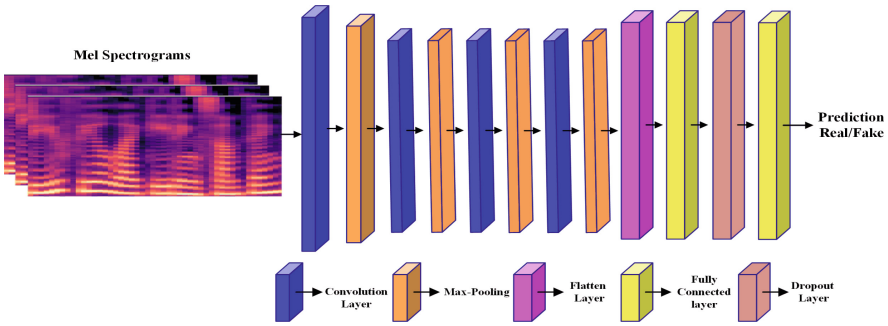


Fig. 3. Proposed Custom-Convolutional Neural Network

3.3 Transfer Learning Approach

Transfer learning is a machine learning technique in which CNNs that have been trained for one task are utilized as the foundation for a model on a different task. We can initialize the weights by employing a pre-trained network that has been trained on large-labeled datasets, such as public image datasets, etc., rather than starting the training process from the very beginning by randomly assigning values to the weights. Pre-trained models trained on ImageNet have demonstrated strong performance and generalization capabilities on various visual recognition tasks. By leveraging a pre-trained model from ImageNet, it can be beneficial to learn rich features representations from a large scale image dataset. In this work, we have used two pre-trained transfer learning models VGG16 [16] and MobileNet [17] for classifying real and fake audios.

Both models' parameters that had been trained using the ImageNet dataset were frozen and loaded in the initial few layers when they were first created [16–18]. The final classification task includes the addition of a flatten layer, which is then followed by a fully connected layer with 256 neurons and ReLU activation, a dropout layer with a dropout ratio of 50%, and a dense layer with 512 neurons and a SoftMax activation function for the final classification task are added. The VGG16 and MobileNet networks were fine-tuned using transfer learning. This enabled efficient learning with a smaller dataset.

4 Experimental Results and Discussion

For the detection of real and fake audios, several experiments are performed using cCNN, VGG16 and MobileNet trained over the sub-datasets of FoR deepfake dataset, for-norm, for-2 s and for-rerecording. We trained VGG16 and MobileNet end-to-end evaluating the performance of such networks against the proposed cCNN on FoR-combined dataset, a combination of the previous three sub-datasets. The experiments do not explicitly include noise or transmission error scenarios. A detailed description of dataset, experimental setup and results are given in the following sub-sections and then compared with the other methods.

4.1 Fake or Real Dataset (foR)

FoR is an audio deepfake dataset consisting of more than 111,000 real utterances collected from speech recording of humans (all genders) and more than 87,000 fake utterances created from 7 different TTS systems. This dataset is divided into 4 different versions based on pre-processing, for-original, for-norm, for-2 s and for-rerecording:

1. for-original: This dataset consists of 195,541 original utterances collected from different sources and is unbalanced in terms of genders and classes i.e., unequal number of real and fake audio sample.
2. for-norm: It is the normalized version of for-original. The audio files are in WAV format and are normalized to 0 dB FS (decibels relative to full scale). It consists of 69,400 utterances and is balanced in terms of genders and classes.
3. for-2 s: This version is similar to for-norm except all the files are trimmed at 2 s. It contains 17,870 utterances.
4. for-rerecording: This dataset consists of files of for-2 s, that have been re-recoded, simulating a real-world attack. 13,268 utterances are present in this version.

In this paper, three versions of FoR dataset, for-norm, for-2 s and for-rerecording have been used. These datasets have already been divided into training, validation and testing by the authors [6] and were used as such in the evaluation process. The total samples present in the dataset and number of samples used for the experiment analysis are given in Table 1.

Additionally, we combined these three sub-datasets and named it as FoR-combined, for extensive experimental analysis. We performed data augmentation on FoR-combined by the introduction of various modifications like height and width shift ranges of 0.2,

zoom range of 0.2, horizontal flip, rotation ranges of 30 degrees, and shear range of 0.2. Table 2 shows the number of samples in the FoR dataset. After the data augmentation, the training data was increased to 41,870.

Table 1. Utterances in three sub-datasets of FoR dataset

Dataset		Total samples		Samples considered	
		Fake	Real	Fake	Real
for-norm	**Train**	26,927	26,941	12,015	12,015
	Val	5398	5400	5398	5400
	Test	2370	2264	2370	2264
for-2 s	**Train**	5104	5104	5104	5104
	Val	1143	1101	1143	1101
	Test	408	408	408	408
for-recording	**Train**	6978	6978	6978	6978
	Val	1413	1413	1413	1413
	Test	544	544	544	544

Table 2. Utterances present in FoR-combined dataset.

	Total no of samples considered	
	Fake	Real
Train	24,097	24,097
Val	7,955	7,955
Test	5,703	5,703

4.2 Experimental Setup

The hardware setup is given in Table 3. Mel spectrograms were fed as an input to all models. For all models the batch size was kept to 32. The models were trained with two different sets of epochs, 20 and 50. For the optimization, Adam optimizer is used, cross entropy as loss function and learning rate for cCNN is fixed to 0.0001 and to 0.001 for VGG16 and MobileNet. The models' performance was assessed in terms of their accuracy throughout training, validation, and testing. Accuracy metric measures the proportion of correctly classified samples among all examples in the dataset. To enable model comparison, we present the results for each classification algorithm in tabular form.

Table 3. Hardware Specifications.

CPU	AMD Ryzen 7 5800X 8-Core 16-Thread Processor 3.80 GHz
GPU	Nvidia G-force RTX 2060 (12 GB)
RAM	16 GB
Hard disk	2 TB SSD

4.3 Experiments Using Proposed cCNN

The proposed cCNN was trained and tested on for-norm, for-2 s, for-rerecording, and for-combined. Table 4 shows the testing accuracy of all the considered datasets. With training epochs of 50, cCNN achieved the highest value of accuracy 97.23% on FoR-combined dataset and 96.32% on for-norm dataset. Furthermore, for-rerecording obtains lower accuracy compared to other datasets with 91.86% and 93.4% with 20 and 50 epochs respectively. Since for-rerecording simulates a real-world attack, it is the most difficult case and a decrement in accuracy is expected.

4.4 Experiments Using Transfer Learning Models

The testing accuracy of VGG16 and MobileNet are depicted in Table 4, with 20 and 50 epochs. With training epochs of 50 VGG16 achieved an accuracy of 95.32% and 94.60% on FoR-combined and for-norm respectively, likewise MobileNet achieved the highest accuracy for FoR-combined and for-norm with 96.13% and 95.18% respectively. However, MobileNet performed better than VGG16 on every dataset. VGG16 achieved the accuracy of 89.8% for for-rerecording with 20 training epochs while MobileNet achieved slightly better accuracy of 90.1% than VGG16 on same configuration and dataset.

Table 4. Performance of the presented models.

Dataset	Custom Models					
	cCNN		VGG 16		Mobile Net	
	20 Epochs %	50 Epochs %	20 Epochs %	50 Epochs %	20 Epochs %	50 Epochs %
for Norm	93.86	96.32	92	94.6	92.8	95.1
for 2 s	92.4	94.1	90.1	92.7	90.9	92.9
for rerecording	91.86	93.4	89.8	91.61	90.1	92.4
FoR-combined	95.8	97.32	93.4	95.32	93.8	96.13

From Table 4, it can be clearly seen that among all datasets, for-rerecording achieved lower results, since it mimics a real-world attack, and is composed of less sample respect

the others. Deep learning models require large amount data for training and learning the data patterns and thus make accurate predictions. FoR-combined achieved the highest accuracies as it is composed of a large number of samples, providing more diverse and representative data for the model to learn from, reducing overfitting, and improving generalization. Overall, the proposed cCNN performed better than the pre-trained models, VGG16 and MobileNet on every dataset used. This demonstrates that a lower number of levels in the network increases the robustness of the model, and it is better suited for the task.

4.5 Experiments Using VGG16 and MobileNet

VGG16 and MobileNet were trained end-to-end using FoR-combined dataset with data augmentation, so that these models could potentially benefit from learning features relevant to the new dataset from the ground up. VGG16 and MobileNet achieved 96.24% and 97.18% of testing accuracy respectively, as shown in Table 5. Also, the EER (Equal Error Rate) is reported in Table 5.

Table 5. Performance of models trained end-to-end.

Models	Accuracy	EER
VGG16	96.24%	0.0376
MobileNet	97.18%	0.0202

On comparing the results of FoR-combined from Table 4 and Table 5, it can be concluded that such models performed marginally better than the pre-trained models, while the proposed cCNN outperforms all the models when trained with 50 epochs.

4.6 Benchmarking

To the best of our knowledge, the considered three FoR sub-datasets, consisting of an increasing level of difficulty due to various post-processing applied to the audio files, have not been the subject of any previous research using CNN. The authors in [13] employed various machine learning algorithms trained on three subsets of FoR datasets, using hand-crafted features. The highest average accuracy of 93% was achieved by the machine learning model eXtreme Gradient Boosting (XGB). Therefore, we also calculated the average accuracy of the presented models for the comparative analysis of the two works.

The effectiveness of the proposed cCNN can be seen in Table 6, obtaining an average boost of 1.60% in the accuracy respect to the state of the art, establishing a new benchmark for the three subsets, for-norm, for-2-seceond and for-rerecording datasets.

Table 6. Comparison with the state-of art

Study	Models	Accuracy %
[13]	XGB	93
Proposed Models	cCNN	94.60
	Pre-trained VGG16	92.97
	Pre-trained MobileNet	93.49

5 Conclusion

The trustworthiness of audio data is crucial because it serves as an essential tool for strengthening security against spoofing and fraud. In this paper, we proposed a Custom Convolution Neural Network to distinguish a fake audio from an original one demonstrating its validity on three versions of FoR datasets, for-norm, for-2-s and for-rerecording datasets and on a for-combined dataset, a combination of the three sub-datasets. All the audio files have been converted to mel spectrograms and used as input to the proposed cCNN, and to the two pretrained transfer learning models, VGG16 and MobileNet. Such networks have also been employed end-to-end on FoR-combined dataset using data augmentation. All the models achieved high accuracy when trained on FoR-combined and lower accuracy when trained on for-rerecording. VGG16 and MobileNet performed slightly better when trained end-to-end. cCNN achieves an accuracy of 97.23% on the combined dataset, performing better than the other considered methods, demonstrating its validity. In the future, we plan to use features fusion and a continual leaning approach for the detection of audio deepfakes to increase robustness and generalization.

Acknowledgements. This study has been partially supported by SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union – NextGenerationEU and Sapienza University of Rome project 2022–2024 "EV2" (003 009 22).

References

1. Masood, M., Nawaz, M., Malik, K.M., Javed, A., Irtaza, A., Malik, H.: Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward. Appl. Intell. **53**(4), 3974–4026 (2023)
2. Akhtar, Z.: Deepfakes generation and detection: a short survey. J. Imaging **9**(1), 18 (2023)
3. Malik, K.M., Malik, H., Baumann, R.: Towards vulnerability analysis of voice-driven interfaces and countermeasures for replay attacks. In: 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 523–528. IEEE (2019)
4. Khanjani, Z., Watson, G., Janeja, V.P.: Audio deepfakes: a survey. Front. Big Data **5**, 1001063 (2023). https://doi.org/10.3389/fdata.2022.1001063
5. Aljasem, M., et al.: Secure automatic speaker verification (SASV) system through SM-ALTP features and asymmetric bagging. IEEE Trans. Inf. Forensics Secur. **16**, 3524–3537 (2021)

6. Firc, A., Malinka, K., Hanácek, P.: Deepfakes as a threat to a speaker and facial recognition: an overview of tools and attack vectors. Heliyon **9**(4), e15090 (2023). https://doi.org/10.1016/j.heliyon.2023.e15090
7. Todisco, M., et al.: ASVspoof 2019: future horizons in spoofed and fake audio detection. arXiv preprint arXiv:1904.05441 (2019)
8. Reimao, R., Tzerpos, V.: For: A dataset for synthetic speech detection. In: 2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), pp. 1–10. IEEE (2019)
9. Khan, A., Sohail, A., Zahoora, U., Qureshi, A.S.: A survey of the recent architectures of deep convolutional neural networks. Artif. Intell. Rev. **53**, 5455–5516 (2020)
10. Wang, R., et al.: Deepsonar: towards effective and robust detection of ai-synthesized fake voices. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 1207–1216 (2020)
11. Camacho, S., Ballesteros, D.M., Renza, D.: Fake speech recognition using deep learning. In: Figueroa-García, J.C., Díaz-Gutierrez, Y., Gaona-García, E.E., Orjuela-Cañón, A.D. (eds.) Applied Computer Sciences in Engineering: 8th Workshop on Engineering Applications, WEA 2021, Medellín, Colombia, October 6–8, 2021, Proceedings, pp. 38–48. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-86702-7_4
12. Khochare, J., Joshi, C., Yenarkar, B., Suratkar, S., Kazi, F.: A deep learning framework for audio deepfake detection. Arab. J. Sci. Eng. **47**(3), 3447–3458 (2021). https://doi.org/10.1007/s13369-021-06297-w
13. Iqbal, F., Abbasi, A., Javed, A.R., Jalil, Z., Al-Karaki, J.: Deepfake Audio Detection via Feature Engineering and Machine Learning (2022)
14. Hamza, A., et al.: Deepfake audio detection via MFCC features using machine learning. IEEE Access **10**, 134018–134028 (2022)
15. Guha, S., Das, A., Singh, P.K., Ahmadian, A., Senu, N., Sarkar, R.: Hybrid feature selection method based on harmony search and naked mole-rat algorithms for spoken language identification from audio signals. IEEE Access **8**, 182868–182887 (2020)
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
17. Howard, A.G., et al.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
18. Alabdulmohsin, I., Maennel, H., Keysers, D.: The impact of reinitialization on generalization in convolutional neural networks. arXiv preprint arXiv:2109.00267 2021