Article

Audio Deep Fake Detection with Sonic Sleuth Model

Anfal Sultan Alshehri , Danah Almalki , Somayah Abdullah Albaradei *

Posted Date: 10 September 2024

doi: 10.20944/preprints202409.0744.v1

Keywords: Artificial Intelligence; Deepfake Detection; Machine Learning; Deep Learning; Deepfake Audio; Audio Forensics; Deepfake

Article

Audio Deep Fake Detection with Sonic Sleuth Model

Anfal Alshehri †, **Dana Almalki** †, **Eaman Alharbi** and **Somayah Albaradei** *

Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Saudi Arabia

* Correspondence: salbaradei@kau.edu.sa

† These authors contributed equally to this work.

Abstract: Information dissemination and preservation are crucial for societal progress, especially in the technological age. While technology fosters knowledge sharing, it also risks spreading misinformation. Audio deep fakes, convincingly fabricated audio using Artificial intelligence (AI), exacerbate this issue. We present Sonic Sleuth, an AI model for detecting audio deepfakes. Our approach leverages Deep Learning (DL) for a robust detection model. Meticulous data preprocessing and rigorous experimentation with various models led to the implementation of the most effective solution with a custom CNN model. Our comprehensive testing resulted in a highly accurate model (98.27% accuracy, 0.016 EER) trained on a substantial dataset of real and synthetic audio. In addition to an 84.92% accuracy and 0.085 EER on an external dataset, these results demonstrate Sonic Sleuth's potential as a powerful tool against audio misinformation.

Keywords: artificial intelligence; deepfake detection; machine learning; deep learning; deepfake audio; audio forensics

1. Introduction

The cybersecurity landscape is constantly evolving, and a novel and highly deceptive threat has emerged: deepfakes. Unlike traditional cyberattacks that often necessitate significant technical prowess, deepfakes leverage the power of artificial intelligence (AI) to create hyper-realistic and convincing fabricated audio, video, or text content. This intrinsic characteristic makes them particularly perilous, as they can effortlessly deceive unsuspecting victims, potentially leading to devastating consequences.

The threat posed by deepfakes is significant and multifaceted, impacting individuals, organizations, and societies. Developing robust deepfake detection methods is crucial for mitigating these risks. By harnessing the power of deep learning, we can create effective tools to identify and combat deepfakes, safeguarding the integrity of digital content and maintaining public trust.

This research addresses a critical challenge in the digital age and contributes to the broader efforts of enhancing cybersecurity and information authenticity.

2. Background

2.1. Deepfake Generation

Deepfakes, a blend of the terms "deep learning" and "fake," refer to images, videos, or audio that have been manipulated or created using artificial intelligence. These media can portray both real and fictional individuals and are categorized as a form of synthetic media.[18] Generating Deepfake is accomplished through a generative adversarial network (GAN), which consists of two parts: Convolutional Neural Networks (CNN) and Deconvolutional Neural Networks (DNN), as shown in Figure 1. They aim to produce data that looks just like real data. The CNN is the generator model, while the DNN is the discriminator model. The generator is trained to produce data, and the discriminator is trained to distinguish between data generated by the generator and real data. When the discriminator successfully distinguishes between real and generated data, the generator generates an enhanced version of the data until the discriminator cannot differentiate between the generated and real data.

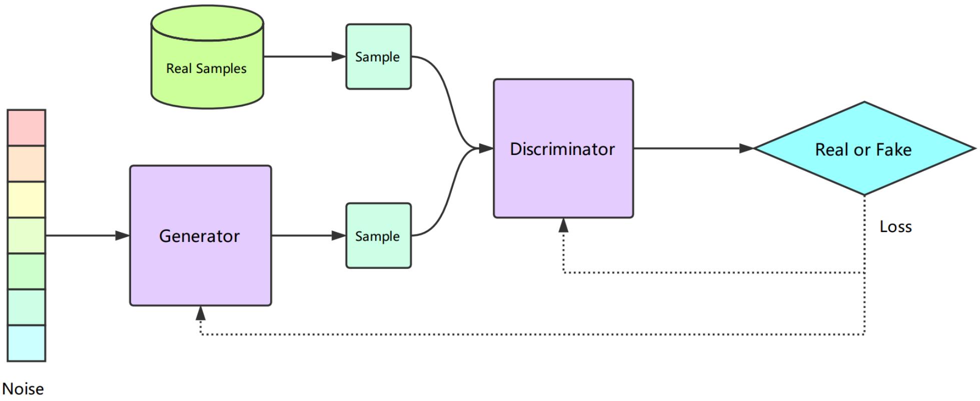

Figure 1. Generative AI structure [12].

2.2. Time Domain and Frequency Domain

The time domain represents a sound wave using a function in time where a graph of such a domain shows the change in the signal value (amplitude) over time [1]. The time domain gives us information about the wave frequency, wavelength, and amplitude. Various calculations can be made to analyze and understand a signal, including Time domain filtering, feature extraction, and signal visualization [1]. Another representation of a signal is the frequency domain, where instead of showing amplitude change over time, it shows how much of a signal lies in a specific frequency over a range of frequencies [2]. Two types of frequency domain graphs are the spectrum and the spectrogram. A spectrum captures the signal's amplitude vs. frequency distribution at a specific point in time, while a spectrogram shows the same information across a period of time; it plots the frequency of a wave over time and uses colors to indicate the amplitude of each frequency, the brighter the color, the higher the amplitude [3].

The Fourier Transform, a powerful mathematical tool, is used to convert signals from the time domain to the frequency domain [2]. This transformation breaks down a signal into the different waves of a constant frequency that make up the whole signal. It provides us with the spectrum of an acoustic wave [2]. The spectrum can then be used to obtain a spectrogram by combining the spectra of multiple overlapping audio segments into a single plot of frequency against time [3].

2.3. Literature Review

Research in the field of Deepfakeaudio detection is a continuous effort to defend against the evolving techniques of generative AI. And just like any AI task, robustness is a main goal of Deepfakeresearch, which requires AI to move to the next level of deep understanding [7].

One way to achieve that depends on the training data of AI models. In the case of Deepfakeaudio detection, a common approach to achieve better generalization of detection is to combine multiple datasets, not only to train your model on both fake and real audio but also to train on data of different nature, such as a dataset that was meant for spoofing attacks or in-the-wild datasets that evaluate your model on realistic audios [15].

The dataset WaveFake [15] is one example where 6 different architectures and a text-to-speech pipeline were used to generate fake audio, addressing the problem of detection models generalization to different Deepfakeaudio generation methods.

A similar approach was that of [9], where the "Attack Agnostic Dataset" was proposed to evaluate the generalization of models. The dataset combined two deepfake datasets and one audio spoofing dataset: WaveFake, FakeAVCeleb, and ASVspoof2019(LA subset), respectively. Spoofing attacks have different elements than Deepfakeaudio since the aim is to deceive Automatic Speech Verification (ASV) systems instead of humans [9].

Table 1. Paper Information

Paper Title	Main Idea	Model	Language	Fakeness Type	Dataset
Arabic Audio Clips: Identification and Discrimination of Authentic Cantillations from Imitations	1- Compare classifiers' performance. 2- Compare recognition with human experts. 3- Authentic reciter identification.	1- Classic 2- Deep learning	Arabic	Imitation	Arabic Diversified Audio
A Compressed Synthetic Speech Detection Method with Compression Feature Embedding	1- Detect synthetic speech in compressed formats. 2- Multi-branch residual network. 3- Evaluate method on ASVspoof. 4- Compare with state-of-the-art.	Deep learning (DNN)	Not specified	Synthetic	ASVspoof
Learning Efficient Representations for Fake Speech Detection	1- Develop efficient fake speech detection models. 2- Ensure accuracy. 3- Adapt to new forms and sources of fake speech.	1- Machine learning (CNN, RNN, SVM, KNN) 2- Deep learning (CNNs, RNNs)	English	Imitation, Synthetic	ASVSpoof, RTVC-Spoof
WaveFake: A Data Set for Audio Deepfake Detection	1- Create a dataset of fake audio. 2- Provide baseline models for detection. 3- Test generalization on data from different techniques. 4- Test on real-life scenarios.	Fake audio synthesizing: MelGAN, Parallel WaveGAN, Multi-band MelGAN, Full-band MelGAN, HiFi-GAN, WaveGlow Baseline detection models: Gaussian Mixture Model (GMM), RawNet2	English, Japanese	Synthetic	WaveFake
Attack Agnostic Dataset: Generalization and Stabilization of Audio Deepfake Detection	1- Analyze and compare model generalization. 2- Combine multiple datasets. 3- Train on inaudible features. 4- Monitor training for stability.	LCNN, XceptionNet, MesoInception, RawNet2, and GMM	English	Synthetic, Imitation	WaveFake, FakeAVCeleb, ASVspoof
Exposing AI-Synthesized Human Voices Using Neural Vocoder Artifacts	1- Utilize the identification of vocoders' artifacts in audio to detect deepfakes. 2- Create a dataset of fake audio using six vocoders.	Detection model: RawNet2 Dataset construction: WaveNet, WaveRNN, Mel-GAN, Parallel WaveGAN, WaveGrad, DiffWave	English	Synthetic	LibriVoc, WaveFake, and ASVspoof2019

Spoofing audio also contains distortion and background noises that aim to confuse ASV systems. These subtle differences can be useful for increasing a model's accuracy and ability to generalize to new real-life data [9].

Additionally, [9] investigated the effect of different front-ends (features) on Deepfakeaudio detection results. This came from the idea that Deepfakeaudio generation models pay attention to the details and features of the audio that are within the hearing scope of humans; therefore, features that are not audible to humans, i.e., of the higher frequency range, might be a giveaway that distinguishes synthesized audio from genuine [9]. Architectures trained on Linear frequency cepstral coefficients (LFCC) front–end performed better than those trained on features that are within the human hearing scope, such as mel–frequency cepstral coefficients (MFCC) and spectrogram–based features [9].

Another approach feature-wise is taking advantage of vocoders as in [10], a neural network that takes the features of an acoustic wave, e.g., Mel-spectrogram, as an input and outputs a waveform [11]. Since vocoders are commonly the last stage in Speech Synthesis frameworks, identifying traces of vocoder's artifacts can help detect Deepfakeaudios [10]. This is possible by training a model to identify the vocoder's artifacts in audio before going through a second model that detects audio Deepfake[10]. **??** provides a summary of the main ideas of the related research

3. Approach

This section outlines our approach to developing the audio deepfake detection models, covering data acquisition, preprocessing, feature extraction, and training.

Existing methods often use limited or specialized datasets, which may not fully capture the diversity of real-world audio manipulations. Our approach, however, utilizes a broad range of datasets to encompass various audio types and manipulation techniques, enhancing our model's detection capabilities.

For feature extraction, we employ techniques like Short-Time Power Spectrum and Constant-Q Transform (CQT). These methods effectively analyze different frequency ranges, ensuring both high and low frequencies are thoroughly examined, which improves the detection of subtle audio manipulations.Figure 2 illustrates the overall approach for detecting deepfakes.

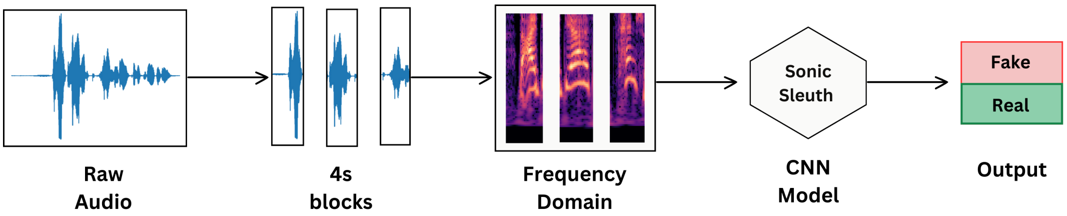

Figure 2. Deepfake Detection Approach

3.1. Datasets

The dataset is a crucial element in deep learning, necessitating a thoughtful and informed selection process that considers the specific task and model requirements. For deepfake audio detection, two main types of data are required: generated audio (fake) and human voice audio (real). We trained our CNN model on three datasets: ASVspoof, In the Wild, and FakeAVCeleb, which combine both types and a total of 70,000 samples. And to ensure the generalization of our model, we further tested it on the Fake-Or-Real Dataset.

- **ASVspoof2019** ASVspoof (Automatic Speaker Verification Spoofing and Countermeasures) is an international challenge focusing on spoofing detection in automatic speaker verification systems [4]. The ASVspoof2019 dataset contains three sub-datasets: Logical access, Physical access, and Speech deepfake. Each of these datasets was created using different techniques depending on the task, like text-to-speech (TTS) and voice conversion (VC) algorithms [4]. We will be utilizing the train subset of the logical access dataset in our experiment.

- **In the Wild** is a dataset containing fake and real audio from 58 celebrities and politicians sourced from publicly available media. It was designed to evaluate deepfake detection models, particularly their ability to generalize to real-world data. [5]
- **FakeAVCeleb** contains both deepfake and real audio and videos of celebrities. The fake audio was created using a text-to-speech service followed by manipulation with a voice cloning tool to mimic the celebrity's voice [6]. We extracted the audio as WAV files from each video.
- **The Fake-or-Real (FoR) Dataset** comprises 111,000 files of real speech and 87,000 files of fake speech. It encompasses both MP3 and WAV file formats. offers four distinct versions to suit various needs. The 'for-original' files are in their original state, while the 'for-norm' version with normalization. 'For-2sec' is shortened to 2 seconds, while 'for-rerec' simulates re-recorded data, depicting deepfake from a phone call scenario [13].

Combining the first three diverse datasets for training and reserving the last one for testing, we aim to achieve a comprehensive and varied dataset for our task. Further details are illustrated in Table 2.

Table 2. Datasets for model development

Name	Size		Length	Sample Rate	File Format	URL
	real	fake				
ASVspoof2019	2,580 files	22,800 files	Avg. 3 s	-	flac	Link
'In-the-Wild'	19,963 files	11,816 files	Avg. 4.s	16 kHz	WAV	Link
FakeAVCeleb	10,209 files	11,357 files	Avg. 5 s	-	MP4	Link
The Fake-or-Real	111,000 files	87,000 files	(1 - 20)s	-	WAV / MP3	Link

3.2. Data Preprocessing

Several essential preprocessing steps are applied to our datasets to ensure uniformity and compatibility of the audio data with our machine-learning models. including:

- First, any silence in the audio was trimmed to enable the model to focus on speech parts of the audio.
- Second, all audio clips are trimmed to a standardized length of 4 seconds or padded by repeating the audio. To avoid silence that could potentially bias the model's learning.
- Third, the audio data is downsampled to a rate of 16 kHz. Downsampling reduces the data size and computational load without significantly compromising the audio signal's quality or essential characteristics.
- Lastly, the audio is converted to a mono channel using channel averaging. Stereo audio, which uses two channels (left and right), is averaged into a single channel, resulting in a mono audio signal.

These conversions simplify the data and ensure a uniform input format for the machine-learning models, which is particularly important when dealing with diverse audio sources.

3.3. Feature Extraction

In audio processing, rather than working with raw audio files, a common practice in AI is to convert audio into a visual representation, such as a spectrogram. This conversion enables the use of image classification models, like Convolutional Neural Networks (CNNs), for audio classification tasks. To accurately capture the essential characteristics of audio signals, we will explore two main types of feature extraction techniques: Short-Time Power Spectrum and Constant-Q Transform (CQT). These techniques are crucial for analyzing different frequency ranges effectively, ensuring that both high and low frequencies are captured for comprehensive analysis.

doi:10.20944/preprints202409.0744.v1

3.3.1. Short-Time Power Spectrum

Short-time power spectrum refers to the representation of the power distribution of a signal over short time intervals. It is a technique commonly used in signal processing and analysis to examine the frequency content and changes in a signal over time. [14]

Our chosen feature extraction techniques, MFCC (Mel-frequency cepstral coefficients) and LFCC (linear frequency cepstral coefficients), are derived from a signal's short-time power spectrum and provide compact representations of its spectral characteristics.

MFCC is a widely used feature set in automatic speech recognition systems[16]. It is based on the human auditory system's response to sound and is particularly effective in capturing the characteristics of speech signals. LFCC is a similar technique that has been shown to outperform MFCC in specific applications, such as deepfake detection [15]. Both techniques are used to extract relevant information from audio signals for further analysis and processing.

3.3.2. Constant-Q Transform

The Constant-Q Transform (CQT) is another feature extraction technique that provides a time-frequency representation of a signal. Unlike the Short-time power spectrum, which has a fixed frequency resolution, CQT offers a logarithmic frequency scale. This means that CQT provides higher frequency resolution at lower frequencies and better time resolution at higher frequencies, which is particularly useful in analyzing music and other signals with a wide frequency range.

CQT is particularly advantageous when dealing with signals where pitch perception is crucial, as it mirrors the human auditory system's logarithmic frequency response. This makes CQT an effective technique in applications like music analysis, instrument identification, and speech processing where capturing the harmonic structure of audio signals is important.[17]

3.4. Sonic Sleuth Structure and Training Details

We employ a Convolutional Neural Network (CNN) to classify processed audio samples into real or synthesized audio. The structure includes 3 convolutional layers, each followed by a max-pooling layer; after the convolutional layers, the output is flattened and passed through two fully connected (dense) layers and a 10% dropout rate to prevent overfitting. The final dense layer has a single output unit with a sigmoid activation function, providing a probability score for classifying the input as deepfake. Figure 3 illustrates the architecture of our custom CNN model.

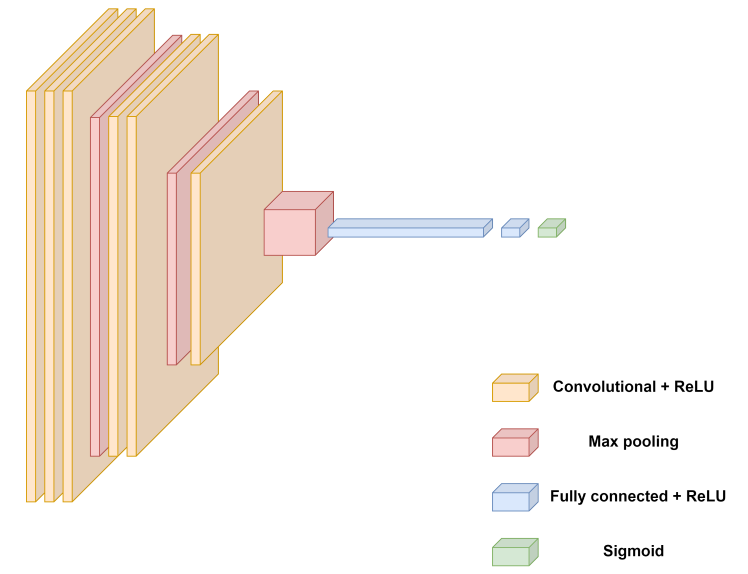

Figure 3. Sonic Sleuth Architecture

For training the deepfake audio detection models, the dataset was split into train, validation, and test subsets with a ratio of 8.5:1:0.5, respectively.

The models were trained for 100 epochs using the following configurations:

1. Early Stopping: An early stopping technique was implemented with a patience of 10 epochs to prevent overfitting. This means the training process would be stopped if the validation loss did not improve for 10 consecutive epochs.
2. Optimizer: The ADAM optimizer was used for updating the model's weights during training.
3. Loss Function: Binary cross-entropy loss was used as the loss function, which is suitable for binary classification problems.
4. Class Weights: Class weights were calculated and applied during the training process to handle the class imbalance in the dataset. Class weights adjust the importance of each class during the loss calculation, helping the model learn from imbalanced datasets more effectively while avoiding bias.

By incorporating these techniques and configurations, the model was trained to learn discriminative features from the audio spectrograms and classify them as real or deepfake with improved performance and generalization to new, unseen data.

3.5. Evaluation Metrics

Equal Error Rate (EER) and accuracy are two commonly used metrics for evaluating the performance of classification models, especially in binary classification tasks. EER represents the point on the Receiver Operating Characteristic (ROC) curve where the false positive rate (FPR) equals the false negative rate (FNR). It is a helpful metric for assessing the balance between the two types of errors, making it the standard metric for evaluating verification systems. A lower EER value indicates better performance, with 0 indicating perfect classification.

Accuracy is a more straightforward metric that measures the proportion of correctly classified instances out of the total number of instances. While accuracy is easy to interpret and widely used, it can be misleading, especially in imbalanced datasets where one class dominates the other. We avoid this mistake in our implementation by ensuring a balanced test dataset and including class weights during training.

In addition to EER and accuracy, the F1 score is another important metric that combines both precision and recall into a single measure. The F1 score is the harmonic mean of precision (the proportion of true positives among the instances classified as positive) and recall (the proportion of true positives among the actual positives). This metric is particularly useful in scenarios where both false positives and false negatives are important, providing a balance between them. A higher F1 score indicates better model performance, especially in cases of class imbalance where precision and recall are critical.

4. Results and Discussion

To accurately represent the results of our experiments, we used several evaluation metrics, including balanced accuracy, F1 score, and Equal Error Rate (EER).

Table 3 presents the results of testing our models on the test subset of our dataset. LFCC demonstrated the best performance across all metrics, achieving an EER of 0.0160, an Accuracy of 98.27%, and an F1 Score of 98.65%. MFCC also performed well, with an EER of 0.0185, an Accuracy of 98.04%, and an F1 Score of 98.45%. In contrast, CQT exhibited the lowest performance, with an EER of 0.0757, an Accuracy of 94.15%, and an F1 Score of 95.76%. These results indicate that LFCC and MFCC are superior feature extraction methods for this specific classification task, providing higher accuracy and lower error rates compared to CQT.

Table 3. Model Performance Metrics

Feature	EER	Accuracy	F1
CQT	0.0757	94.15%	95.76%
LFCC	0.0160	98.27%	98.65%
MFCC	0.0185	98.04%	98.45%

4.1. Test on External Dataset

To assess the generalizability of our models, we tested them on an external dataset, Fake-or-Real (FoR). The testing was conducted on a balanced subset of 2116 samples of 4-second audio. The performance metrics are reported in Table 4.

Table 4. Model Performance Metrics on External Dataset

Feature	EER	Accuracy	F1
CQT	0.0942	82.51%	83.19%
LFCC	0.1718	75.28%	72.63%
MFCC	0.3165	61.24%	54.94%

While the performance of the models has decreased compared to the initial evaluation, it's noteworthy that CQT, which previously had the lowest accuracy among the three methods, now showcases improved performance. With an EER of 0.0942, an Accuracy of 82.51%, and an F1 Score of 83.19%, CQT has made significant strides, indicating its viability as a feature extraction method even in diverse datasets.

Similarly, LFCC, despite experiencing a decline in performance, continues to exhibit its utility in real-world scenarios. LFCC, with an EER of 0.1718, an Accuracy of 75.28%, and an F1 Score of 72.63%, maintains its effectiveness in capturing relevant audio features. Meanwhile, MFCC, with an EER of 0.3165, an Accuracy of 61.24%, and an F1 Score of 54.94%, appears not to be the best for real-life applications due to weak generalizability.

4.2. Ensemble Approach

In machine learning, the ensemble model approach combines the output of multiple models by combining the output probabilities (percentages) from three models, each contributing up to 100%. The final decision is based on the combined score, normalized by dividing the total by 300 to yield a final classification trained for the same task, which can reduce error.

Table 5 showcases the results of the ensemble approach that combines CQT and LFCC models.

Table 5. Model Performance Metrics on External Dataset (Ensemble Approach)

Feature	EER	Accuracy	F1
CQT and LFCC	0.0851	84.92%	84.73%

The combination of CQT and LFCC on the external dataset presents promising results, with an Equal Error Rate (EER) of 0.0851, an Accuracy of 84.92%, and an F1 Score of 84.73%. This joint approach showcases slight improvement compared to the individual methods, indicating the complementary nature of CQT and LFCC in capturing relevant audio features. These results provide valuable insights for further exploration and optimization, highlighting the potential for synergy between different feature extraction methods to achieve superior performance across diverse datasets and application contexts.

5. Conclusion

In conclusion, our research centered on the development of an AI model named Sonic Sleuth for the detection of audio deepfakes. Utilizing a custom convolutional neural network (CNN), we trained our model on three diverse datasets: ASVspoof, In the Wild, and FakeAVCeleb, amassing a total of 78,725 audio samples. To prepare the data, we employed feature extraction techniques like Linear Frequency Cepstral Coefficients (LFCC), Mel-Frequency Cepstral Coefficients (MFCC), and Constant-Q Transform (CQT) to convert audio signals into spectrograms for detailed analysis.

Our study demonstrated that LFCC performed optimally on the training dataset, achieving an EER of 0.0160 and an accuracy of 98.27%. Conversely, CQT showed superior performance on the external dataset, indicating better generalization with an EER of 0.0942 and an accuracy of 82.51%. Moreover, the ensemble approach combining CQT and LFCC further improved performance, yielding an EER of 0.0851 and an accuracy of 84.92%.

For future work, we propose further exploration of various ensemble methods to boost performance. Integrating a broader range of features could help identify the most complementary combinations, thereby enhancing the model's performance and generalization capabilities even further.

Author Contributions: Conceptualization, A.A., D.A.; methodology, A.A., D.A.; software, A.A., D.A.; validation, E.A., S.A.; formal analysis, A.A., D.A.; investigation, A.A., D.A.; resources, A.A., D.A.; data curation, A.A., D.A.; writing—original draft preparation, A.A., D.A.; writing—review and editing, E.A., S.A.; visualization, A.A., D.A.; supervision, E.A., S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was conducted without external financial support. The project relied on the advanced computational tools, software, and datasets that were integral to the development and testing of the Sonic Sleuth model. The research team's dedication, combined with access to high-quality resources such as deep learning frameworks and substantial datasets, was pivotal in achieving the study's goals. The institution supported us by providing the necessary infrastructure and opportunities to demonstrate our capabilities, which played a crucial role in the successful completion of this research.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: This work was written by Anfal Alshehri and Dana Almalki, under the supervision of Dr.Eaman Alharbiand and Dr.Somayah Albaradei. We are grateful for the guidance and support provided throughout this project.

References

1. Camastra, F.; Vinciarelli, A. *Machine Learning for Audio, Image and Video Analysis: Theory and Applications*; Springer: London, UK, 2015.
2. Tenoudji, F.C. *Analog and Digital Signal Analysis: From Basics to Applications*; Springer International Publishing: Cham, Switzerland, 2018.
3. Natsiou, A.; O'Leary, S. Audio Representations for Deep Learning in Sound Synthesis: A Review. *arXiv*, 2022, *cs.SD*. Available online: https://arxiv.org/abs/2201.02490 (accessed on Day Month Year).
4. Wang, X.; Yamagishi, J.; Todisco, M.; Delgado, H.; Nautsch, A.; Evans, N.; Sahidullah, M.; Vestman, V.; Kinnunen, T.; Lee, K.A.; Juvela, L.; Alku, P.; Peng, Y.-H.; Hwang, H.-T.; Tsao, Y.; Wang, H.-M.; Le Maguer, S.; Becker, M.; Henderson, F.; Clark, R.; Zhang, Y.; Jia, Q.; Onuma, K.; Mushika, K.; Kaneda, T.; Jiang, Y.; Liu, L.-J.; Wu, Y.-C.; Huang, W.-C.; Toda, T.; Tanaka, K.; Kameoka, H.; Steiner, I.; Matrouf, D.; Bonastre, J.-F.; Govender, A.; Ronanki, S.; Zhang, J.-X.; Ling, Z.-H. ASVspoof 2019: A Large-Scale Public Database of Synthesized, Converted and Replayed Speech. *arXiv*, 2020, *eess.AS*. Available online: https://arxiv.org/abs/1911.01601 (accessed on Day Month Year).
5. Müller, N.M.; Czempin, P.; Dieckmann, F.; Froghyar, A.; Böttinger, K. Does Audio Deepfake Detection Generalize? *Interspeech*, 2022.
6. Khalid, H.; Tariq, S.; Kim, M.; Woo, S.S. FakeAVCeleb: A Novel Audio-Video Multimodal Deepfake Dataset. In Proceedings of the Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2), 2021. Available online: https://openreview.net/forum?id=TAXFsg6ZaOl (accessed on Day Month Year).

7. Marcus, G. The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence. *CoRR*, 2020, *abs/2002.06177*. Available online: https://arxiv.org/abs/2002.06177 (accessed on Day Month Year).

15. Frank, J.; Schönherr, L. WaveFake: A Data Set to Facilitate Audio Deepfake Detection. *CoRR*, 2021, *abs/2111.02813*. Available online: https://arxiv.org/abs/2111.02813 (accessed on Day Month Year).

9. Kawa, P.; Plata, M.; Syga, P. Attack Agnostic Dataset: Towards Generalization and Stabilization of Audio DeepFake Detection. In Proceedings of Interspeech 2022, ISCA, September 2022. DOI: 10.21437/interspeech.2022-10078.

10. Sun, C.; Jia, S.; Hou, S.; AlBadawy, E.; Lyu, S. Exposing AI-Synthesized Human Voices Using Neural Vocoder Artifacts. *arXiv*, 2023, *cs.SD*. Available online: https://arxiv.org/abs/2302.09198 (accessed on Day Month Year).

11. Zhang, C.; Zhang, C.; Zheng, S.; Zhang, M.; Qamar, M.; Bae, S.-H.; Kweon, I.S. A Survey on Audio Diffusion Models: Text To Speech Synthesis and Enhancement in Generative AI. *arXiv*, 2023, *cs.SD*. Available online: https://arxiv.org/abs/2303.13336 (accessed on Day Month Year).

12. Gu, Y.; Chen, Q.; Liu, K.; Xie, L.; Kang, C. GAN-based Model for Residential Load Generation Considering Typical Consumption Patterns. In Proceedings of ISGT 2019, IEEE, November 2018. DOI: 10.1109/ISGT.2019.8791575.

13. Abdeldayem, M. The Fake-or-Real Dataset. *Kaggle Dataset*, 2022. Available online: https://www.kaggle.com/datasets/mohammedabdeldayem/the-fake-or-real-dataset (accessed on Day Month Year).

14. Sahidullah, M.; Kinnunen, T.; Hanilçi, C. A Comparison of Features for Synthetic Speech Detection. *Interspeech 2015*; 2015. Available online: https://doi.org/10.21437/Interspeech.2015-472 (accessed on Day Month Year).

15. Frank, J.; Schönherr, L. WaveFake: A Data Set to Facilitate Audio Deepfake Detection. *CoRR*; 2021, Volume abs/2111.02813. Available online: https://arxiv.org/abs/2111.02813 (accessed on Day Month Year).

16. Zheng, F.; Zhang, G. Integrating the energy information into MFCC. *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP 2000)*; 2000, Volume 1, pp. 389-392. Available online: https://doi.org/10.21437/ICSLP.2000-96 (accessed on Day Month Year).

17. Todisco, M.; Delgado, H.; Evans, N. Constant Q Cepstral Coefficients: A Spoofing Countermeasure for Automatic Speaker Verification. *Computer Speech & Language* 2017, *45*, 516–535. Available online: https://doi.org/10.1016/j.csl.2017.01.001 (accessed on 14 February 2020).

18. Deepfakes (a portmanteau of "deep learning" and "fake"). Images, videos, or audio edited or generated using artificial intelligence tools. *Synthetic Media*, 2023. Available online: https://en.wikipedia.org/wiki/Deepfake (accessed on Day Month Year).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.