IEEE Forum on Research and Technologies for Society and Industry (RTSI), Paris, France, August 2022

Detecting Replay Attack on Voice-Controlled Systems using Small Neural Networks

Nadeen Ahmed
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
g00079991@aus.edu

Jowaria Khan
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
g00084343@aus.edu

Nouran Sheta
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
g00080065@aus.edu

Rahma Tarek
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
g00079160@aus.edu

Imran Zualkernan
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
izualkernan@aus.edu

Fadi Aloul
Computer Science and Engineering
American University of Sharjah
Sharjah, UAE
faloul@aus.edu

***Abstract*- Voice-control is becoming a common interface for many consumer IoT systems. Common threats to such systems include impersonation, replay, speech synthesis, and voice conversion attacks. Of these attacks, replay is the easiest to implement where a command is recorded and replayed. This paper explores the development of a lightweight intrusion detection neural network based on a recent command voice replay dataset. A lightweight model based on 1D Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) was proposed. The trained model was compared with baseline models based on Gaussian Mixture Models (GMM) using Constant Q Cepstral Coefficients (CQCC) and Mel-Frequency Cepstral Coefficient (MFCC). The proposed model outperformed the GMM models, and its size was significantly lower making it more feasible for embedded systems implementation.**

Keywords— replay attack, IoT, deep learning, voice-controlled systems, convolutional neural network, audio classification, ReMASC

I. Introduction

Voice interfaces have increasingly been incorporated into home devices. Many smart home appliances today can be voice-controlled to adjust temperature, activate home security systems, or to shop online, for example. While convenient, such voice interfaces expose the home owners to a variety of attacks [1]. These attacks include impersonation, replay, speech synthesis, and voice conversion attacks [2]. Impersonation is when an attacker sounds like a target speaker. Similarly, a replay attack involves an attacker presenting a prerecorded speech sample from the target speaker. Speech synthesis is the process of creating artificial speech and voice conversion tries to covert the speaker's voice to sound like the target's voice. Most smartphones today have smart voice assistants such as Google Assistant, Siri and Cortana that assist users to control their phones or IoT devices. Some threats to these voice-controlled systems (VCSs) include the unlocking of doors, making unauthorized purchases, controlling sensitive home appliances, and transmitting sensitive information [3].

Among the aforementioned attacks, replay attacks are the easiest to implement by using a recording device. The counter measure to this type of attack is hence the ability to tell the original speech signal apart from the signal when it is played back through a playback device. This paper explores how replay attacks can be detected using deep learning techniques. Specifically, the paper explores how a light-weight neural network can be developed to detect such spoofing using a recent dataset of voice commands for replay attacks in realistic scenarios called Realistic Replay Attack Microphone Array Speech Corpus (ReMASC) [1].

II. Related work

Equal Error Rate (EER) is a commonly used metric to measure how good spoof detection systems are. EER is based on the Detection Error Tradeoff (DET) curve where EER is the point on the curve where the False Rejection Rate (FRR) and False Acceptance Rate (FAR) are equal and minimal [4]. The lower the EER, the better the system at identifying spoofing. Most spoofing detection systems consist of two components: feature extraction and classification. Feature extraction consists of determining the relevant features that help distinguish real from spoofed speech while the classification stage constructs a discriminator based on the features.

A. Support Vector Machine (SVM)

Methods relying on iterative adaptive Inverse Filtering (IA-IF) and Linear Frequency Cepstral Coefficients (LFCC) have resulted in 8.32% EER and 22.65% EER respectively [5],[6]. Similarly, Acoustic Ternary Patterns-Gammatone Cepstral Coefficient (ATP-GTCC) used with the multi-class SVM classifier yielded an EER of 0.6% [7], [8]. Lavrentyeva et al. [9] proposed a fusion of SVM i-vector, Light CNN (LCNN) with Fast Fourier Transforms (FFT), CNN with FFT, and Recurrent Neural Networks (RNN). The system scored 3.95% EER and 6.73% EER on the development and evaluation set of the ASVspoof 2017 dataset [10].

B. Convolution Neural Network (CNN)

Parasu et al. created light ResNet with spectrogram features which outperformed CQCC-GMM and Attentive Filtering

Network (AFN) baselines [11], [12]. The model achieved 0.81% EER on ASVspoof 2015. Hyun et al. [13] proposed a multiple points input method to increase the amount of information that could be considered at one time. This method handled the limitation of the low amount of information that is considered at a time in CNN-based models. This methodology also reduced the relative EER by about 44% compared to the baseline. Duraibi et al. [14] developed a new approach that uses MFCC and CQCC as input features, a CNN based front-end feature extractor, and SVM back-end classifier. The experiment conducted on ASVspoof 2017 datasets showed an improvement in the state-of-the-art performance achieving 7.1% EER. Elsaeidy et al. [15] proposed a deep CNN architecture consisting of an input layer, four hidden layers, a global average pooling layer and an output layer that achieved an accuracy of 98.04%, False Positive Rate (FPR) of 5.66%, sensitivity of 97.64%, specificity of 97.6422%, and precision of 97.64% compared to baseline. Another approach using Constant Q Cepstral Coefficients (CQCC) features and a CNN as an input to a LSTM classifier resulted in 7.73% EER on the ASVspoof dataset [16],[17].

C. *Gaussian Mixture Models (GMM)*

Tan et al [19] conducted a survey to explore the limitations and future directions of Presentation Attack Detection (PAD). They found that the GMM were more robust than other speaking modelling approaches. Suthokumar et al. proposed a system that used Spectro-Temporal Modulation Features (STMF) and CQCC to obtain an EER of 7.11% and 0.83% on ASVSpoof 2017 and BTAS 2016 datasets [20], [21]. Pradhan et al. [22] developed an end-to-end system called REVOLT that uses SVM and GMM models and their ensemble to detect replay attacks that intelligently exploits the inherent differences between the spectral characteristics of the original and replayed voice signals. Their proposed system yields best EER of 0.88% and 10.32% respectively in their own dataset and ASV2017 dataset respectively as compared to standard LFCC and MFCC combined with GMM and SVM as a classifier.

D. *Deep Neural Networks (DNN)*

Gomez-Alanis et al. [23] trained DNN based on a loss function that used kernel density estimation (KDE) techniques. Their results outperformed the systems trained using other loss function with EER of 0.82% on ASVspoof 2019. Duraibi et al. created a DNN classifier using hybrid features from Mel-frequency cepstral coefficient (MFCC) and CQCC which outperformed the conventional GMM classifier [24], [25]. The model achieved EERs of 2.68% on development set, 7.65% on evaluation set, and 5.64% on development and evaluation set. Jung et al. [26] detected unrevealed characteristics that reside in a replayed speech by directly inputting spectrograms into an end-to-end DNN without knowledge-based intervention. Their experiments conducted on the ASVspoof 2019 physical access challenge showed promising results, where EER was 2.45 % for the evaluation set.

In summary most previous approaches have used feature extraction like CQCC or MFCC in conjunction with classification techniques like SVM, CNN, DNN, GMM or LSTM to achieve reasonable EER on primarily ASVProof-type datasets.

III. Methodology

A. *Dataset Description*

This paper uses the Realistic Replay Attack Microphone Array Speech Corpus (ReMASC) [1]. Unlike previous datasets, this dataset contains both authentic voice commands and replayed recordings of these commands that were collected in a realistic setting. This dataset has been designed specifically for voice-command replay attacks. Recordings from 50 subjects are included that vary in age, gender, and accents. The dataset contains 132 voice commands in four environments, two indoor, one outdoor, and one in a moving vehicle, with different forms of background noise. The relative positions between speaker and device range from 0.5m to 6m with varying placements and microphone configuration of devices. The corpus contains the core, evaluation, and complete datasets. The complete set has all the data that is in the core set and the evaluation set to allow freedom of splitting the training/test split of the model. The core and evaluation sets have a default training/test split. The data contains 54,712 audio clips out of which 9,240 are genuine and 45,472 are replayed.

B. *Data Preprocessing and Feature Engineering*

All clips were converted to mono, down sampled to 16kHz, and the bit-depth was normalized between -1 and 1. Fig. 1 shows examples of genuine and replay clips.

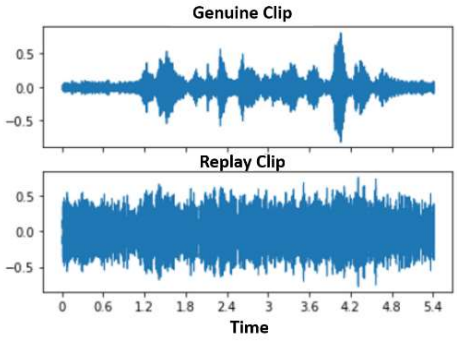

Fig. 1. Time domain signal of the genuine and replay clips

Next, each signal was randomly sampled into 5 samples of 0.5 seconds each. Short-Term Fourier Transform (STFT) was then calculated. An example of STFT of the genuine and replay clips is shown in Fig. 2.

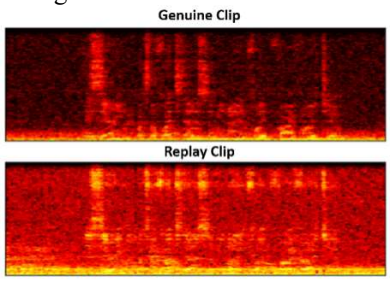

Fig. 2. STFT of the genuine and replay signals

Fig. 3 shows the results of subsequent creation of a Mel spectrogram after applying Mel filters.

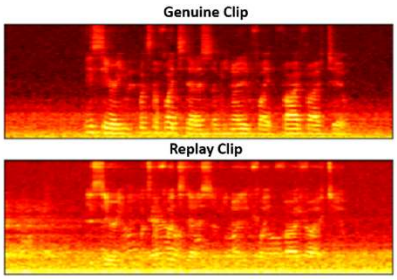

Fig. 3. Mel spectrogram of the audio signals

Finally, a discrete cosine transform was then performed on the logs of the Mel Spectrogram to produce MFCC as shown in Fig. 4.

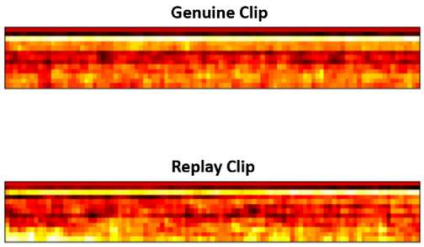

Fig. 4. The Mel Frequency Cepstral Coefficient of the audio signals

C. *Models Used*

The 1D CNN with LSTM (1D CNN-LSTM) model shown in Fig. 5 was used. Table I shows the various layers in more detail. The MFCC representation of the audio signal, which is shown in Fig. 4, was fed into a 1DCNN for feature recognition and the time sequence was being captured by an LSTM. The classifier was a simple DNN. A 1D CNN has lower computational complexity than a 2D CNN and uses more compact configuration [27]. In addition to a 1D CNN, an LSTM was used to support time series sequence prediction [28]. Both CNN and LSTM were used in this model for feature extraction followed by a fully connected network for classification.

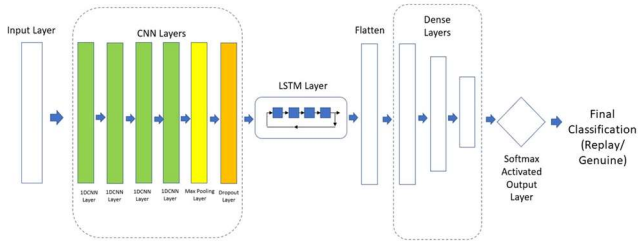

Fig. 5. The 1D CNN-LSTM Model

To be able to evaluate the overall performance of the models, a baseline GMM combined with MFCC and CQCC features was trained on the same dataset [29].

TABLE I. VARIOUS LAYERS IN THE 1DCNN+LSTM MODEL

Layers	Parameters
Con1D	Filter_size = 16, Kernel_size = 3, activation = "relu", strides = 1, padding = "same"
Con1D	Filter_size = 32, Kernel_size = 3, activation = "relu", strides = 1, padding = "same"
Con1D	Filter_size = 64, Kernel_size = 3, activation = "relu", strides = 1, padding = "same"
Con1D	Filter_size = 128, Kernel_size = 3, activation = "relu", strides = 1, padding = "same"
MaxPool1D	Pool size=2
Dropout	Rate=0.5
LSTM	Units=128
Flatten	-
Dense	Units=128, activation= "relu"
Dense	Units=64, activation= "relu"
Dense	Units=2, activation= "relu"

IV. RESULTS

10-Fold cross validation was used to assess the model robustness. The models were evaluated using a variety of metrics including EER [30]. The 10-Fold cross-validation generally resulted in very low standard deviations for most metrics (e.g., 0.006). All models were evaluated on a subset of the development set which was unseen by the model and on an evaluation set also unseen. The various evaluation metrics for each model are shown in the Table I below.

TABLE II. COMPARING GMM WITH 1DCNN+LSTM

Dataset	Metric	GMM		1DCNN +LSTM
		MFCC	***CQCC***	***MFCC***
Development	Accuracy	51%	38%	**83%**
	Precision	60%	57%	**83%**
	Recall	51%	38%	**83%**
	F1 Score	55%	46%	**83%**
	AUC	0.50	0.50	**0.92**
	EER	50%	48.1%	**28.1%**
Evaluation	Accuracy	24%	34%	**81%**
	Precision	71%	75%	**86%**
	Recall	24%	34%	**81%**
	F1 Score	27%	41%	**83%**
	AUC	0.56	0.58	**0.88**
	EER	46.1%	47.3%	**31.9%**

As Table II shows, the 1DCNN+LSTM model using MFCC outperformed the baseline GMM models. The 1DCNN+LSTM model achieved a decent F1-Score of 83% for both the unseen development and evaluation datasets. The EER seemed high for the models but were closer to the best EER reported for this dataset. The GMM obviously did not perform well and there does not seem to be a difference between the usage of MFCC or CQCC in terms of the results.

Since any spoof detection model would typically run on an embedded device with low computational capability, the size of the model is also important. As Table III shows, the 1DCNN+LSTM model was significantly smaller than either version of the GMM model.

As shown in Fig. 6 and Fig. 7, the model training and validation losses and accuracies did not deviate much suggesting that the model did not overfit. Fig. 8 and Fig. 9 show the Receiver Operating Curves (ROC) for the unseen development and evaluation data for the best models. As expected, the Area Under the Curve (AUC) for the development

data was 0.89 and a bit better than the AUC of the evaluation data which was 0.80.

TABLE III. COMPARISON BETWEEN GMM AND 1DCNN+LSTM SIZE

Model	Features	Size in MB	F1 Score	
			Development	*Evaluation*
GMM	MFCC	9.75	0.53	0.27
	CQCC	300.82	0.62	0.71
1DCNN+LSTM	MFCC	**0.80**	**0.8**	**0.83**

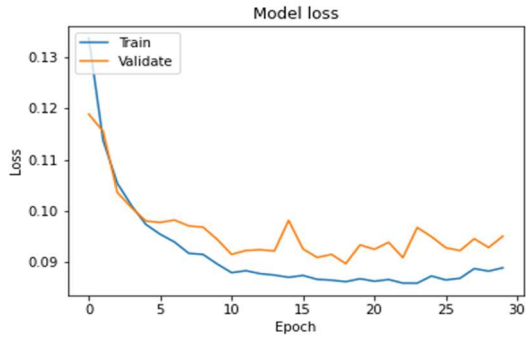

Fig. 6. Loss Curve for the Training and Validation Data for 1DCNN+LSTM Model

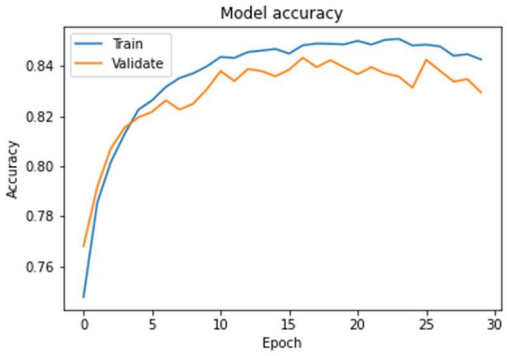

Fig. 7. Accuracy Curve for the Training and Validation Data for 1DCNN+LSTM Model

Since the dataset was imbalanced, Synthetic Minority Over-sampling Technique (SMOTE) was used to see if the performance would improve [31]. SMOTE is a synthetic minority oversampling technique which statistically increases the number of cases in a dataset by generating new instances from existing minority cases while keeping the majority classes as is. However, using SMOTE did not result in a significant change in the results as it raised accuracies and the F1 scores by around 1%-2%.

Overall, the proposed model achieved higher results on the development set than the evaluation set which showed that it classified attacks in environments similar to the ones it had been trained on better than those on which it had not been trained. To achieve close results between seen and unseen environments, features could be extracted from the complete set instead of only the core set so that it could capture all the different environments available in the dataset. The captured features can then be split into training and testing sets to be passed later to the model. To further improve the model performance, grid search can be conducted to find the most optimal number of filters and features. These hyperparameters can then be used in feature extraction functions like MFCC to capture most of the audio important information. Since the model had only been trained using audio files with mono channel configuration, models that leverage features extracted from the multi-channels of the audio could also be explored. Additionally, latest models like transformers could be explored to detect replay attacks. Finally, the most obvious extension is to use 2D CNNs that have been used in many audio classification tasks.

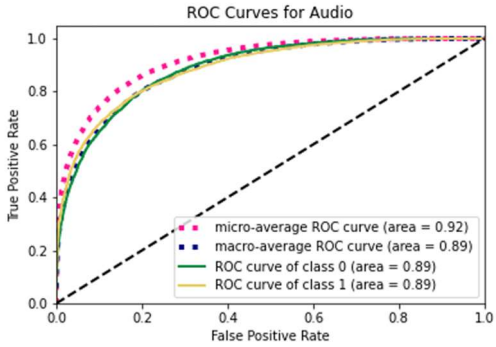

Fig. 8. ROC Curve for the Development Data for 1DCNN+LSTM Model

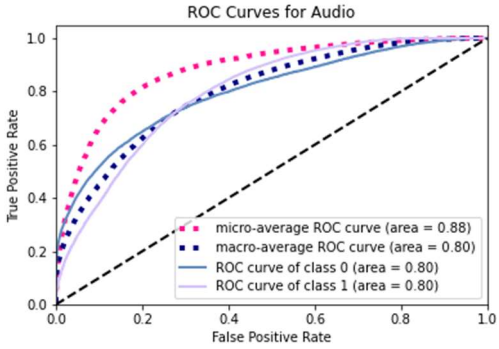

Fig. 9. ROC Curve for the Evaluation Data for 1DCNN+LSTM Model

V. CONCLUSION

The number of voice-controlled systems have been increasing as IoT devices become more common. Consequently, there are also ever-increasing security threats associated with such systems. These threats include replay attacks, self-triggered attacks, hidden voice commands and audio adversarial attacks. This paper has shown that it is possible to derive fairly small neural networks with less than 180k parameters to detect such attacks making these models feasible for embedded devices.

REFERENCES

[1] Y. Gong, J. Yang, J. Huber, M. MacKnight, and C. Poellabauer, "ReMASC: Realistic Replay Attack Corpus for Voice Controlled Systems," Sep. 2019, pp. 2355–2359. doi: 10.21437/Interspeech.2019-1541.

[2] M. Singh and D. Pati, "Chapter 2 - Replay attack detection using excitation source and system features," in *Advances in Ubiquitous*

Computing, A. Neustein, Ed. Academic Press, 2020, pp. 17–44. doi: 10.1016/B978-0-12-816801-1.00002-5.

[3] Y. Gong and C. Poellabauer, "Protecting Voice Controlled Systems Using Sound Source Identification Based on Acoustic Cues," *2018 27th Int. Conf. Comput. Commun. Netw. ICCCN*, pp. 1–9, Jul. 2018, doi: 10.1109/ICCCN.2018.8487334.

[4] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," National Inst of Standards and Technology Gaithersburg MD, 1997.

[5] K. P. Bharath and M. Rajesh Kumar, "New replay attack detection using iterative adaptive inverse filtering and high frequency band," *Expert Syst. Appl.*, vol. 195, p. 116597, Jun. 2022, doi: 10.1016/j.eswa.2022.116597.

[6] S. Jana, V. S. Yashwanth, K. V. N. Dheeraj, S. Balaji, K. P. Bharath, and M. Rajesh Kumar, "Replay Attack Detection for Speaker Verification Using Different Features Level Fusion System," in *2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, Nov. 2021, pp. 1–5. doi: 10.1109/i-PACT52855.2021.9696686.

[7] K. M. Malik, A. Javed, H. Malik, and A. Irtaza, "A Light-Weight Replay Detection Framework For Voice Controlled IoT Devices," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 5, pp. 982–996, Aug. 2020, doi: 10.1109/JSTSP.2020.2999828.

[8] D. Meyer, F. Leisch, and K. Hornik, "The support vector machine under test," *Neurocomputing*, vol. 55, no. 1, pp. 169–186, Sep. 2003, doi: 10.1016/S0925-2312(03)00431-4.

[9] G. Lavrentyeva, S. Novoselov, E. Malykh, A. Kozlov, K. Oleg, and V. Shchemelinin, *Audio Replay Attack Detection with Deep Learning Frameworks*. 2017, p. 86. doi: 10.21437/Interspeech.2017-360.

[10] Z. Wu *et al.*, "ASVspoof: The Automatic Speaker Verification Spoofing and Countermeasures Challenge," *IEEE J. Sel. Top. Signal Process.*, vol. 11, no. 4, pp. 588–604, Jun. 2017, doi: 10.1109/JSTSP.2017.2671435.

[11] P. Parasu, J. Epps, K. Sriskandaraja, and G. Suthokumar, "Investigating Light-ResNet Architecture for Spoofing Detection Under Mismatched Conditions," in *Interspeech 2020*, Oct. 2020, pp. 1111–1115. doi: 10.21437/Interspeech.2020-2039.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," arXiv, arXiv:1512.03385, Dec. 2015. doi: 10.48550/arXiv.1512.03385.

[13] S.-H. Yoon and H.-J. Yu, "Multiple Points Input For Convolutional Neural Networks in Replay Attack Detection," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 6444–6448. doi: 10.1109/ICASSP40776.2020.9053303.

[14] S. Duraibi, W. Alhamdani, and F. T. Sheldon, "Voice Feature Learning using Convolutional Neural Networks Designed to Avoid Replay Attacks," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2020, pp. 1845–1851. doi: 10.1109/SSCI47803.2020.9308489.

[15] A. A. Elsaeidy, N. Jagannath, A. G. Sanchis, A. Jamalipour, and K. S. Munasinghe, "Replay Attack Detection in Smart Cities Using Deep Learning," *IEEE Access*, vol. 8, pp. 137825–137837, 2020, doi: 10.1109/ACCESS.2020.3012411.

[16] 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.

[17] L. Huang and J. Zhao, "Audio replay spoofing attack detection using deep learning feature and long-short-term memory recurrent neural network," in *AIIPCC 2021; The Second International Conference on Artificial Intelligence, Information Processing and Cloud Computing*, Jun. 2021, pp. 1–5.

[18] Ç. Süslü, E. Eren, and C. Demiroğlu, "Uncertainty assessment for detection of spoofing attacks to speaker verification systems using a Bayesian approach," *Speech Commun.*, vol. 137, pp. 44–51, Feb. 2022, doi: 10.1016/j.specom.2021.12.003.

[19] C. B. Tan *et al.*, "A survey on presentation attack detection for automatic speaker verification systems: State-of-the-art, taxonomy, issues and future direction," *Multimed. Tools Appl.*, vol. 80, no. 21, pp. 32725–32762, Sep. 2021, doi: 10.1007/s11042-021-11235-x.

[20] G. Suthokumar, V. Sethu, K. Sriskandaraja, and E. Ambikairajah, "Adversarial Multi-Task Learning for Speaker Normalization in Replay Detection," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 6609–6613. doi: 10.1109/ICASSP40776.2020.9054322.

[21] P. Korshunov *et al.*, "Overview of BTAS 2016 speaker anti-spoofing competition," in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, Sep. 2016, pp. 1–6. doi: 10.1109/BTAS.2016.7791200.

[22] "Combating Replay Attacks Against Voice Assistants | Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies." https://dl.acm.org/doi/abs/10.1145/3351258 (accessed May 09, 2022).

[23] A. Gomez-Alanis, J. A. Gonzalez-Lopez, and A. M. Peinado, "A Kernel Density Estimation Based Loss Function and its Application to ASV-Spoofing Detection," *IEEE Access*, vol. 8, pp. 108530–108543, 2020, doi: 10.1109/ACCESS.2020.3000641.

[24] S. Duraibi, W. Alhamdani, and F. T. Sheldon, "Replay Spoof Attack Detection using Deep Neural Networks for Classification," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, Dec. 2020, pp. 170–174. doi: 10.1109/CSCI51800.2020.00036.

[25] L. Muda, M. Begam, and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques," arXiv, arXiv:1003.4083, Mar. 2010. doi: 10.48550/arXiv.1003.4083.

[26] J. Jung, H. Shim, H.-S. Heo, and H.-J. Yu, "Replay attack detection with complementary high-resolution information using end-to-end DNN for the ASVspoof 2019 Challenge," *ArXiv190410134 Cs Eess*, Jul. 2019, Accessed: May 09, 2022. [Online]. Available: http://arxiv.org/abs/1904.10134

[27] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mech. Syst. Signal Process.*, vol. 151, p. 107398, Apr. 2021, doi: 10.1016/j.ymssp.2020.107398.

[28] J. Brownlee, "CNN Long Short-Term Memory Networks," *Machine Learning Mastery*, Aug. 20, 2017. https://machinelearningmastery.com/cnn-long-short-term-memory-networks/ (accessed May 04, 2022).

[29] "2.1. Gaussian mixture models — scikit-learn 0.15-git documentation." https://scikit-learn.org/0.15/modules/mixture.html#gmm (accessed May 06, 2022).

[30] mino, "Calculate EER from FAR and FRR?," *Cross Validated*, Jun. 30, 2016. https://stats.stackexchange.com/q/221562 (accessed May 06, 2022).

[31] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.