Speech Communication 154 (2023) 102973

Contents lists available at ScienceDirect

Speech Communication

journal homepage: www.elsevier.com/locate/specom

DNN controlled adaptive front-end for replay attack detection systems

Buddhi Wickramasinghe [a,b,*], Eliathamby Ambikairajah [a,b], Vidhyasaharan Sethu [a], Julien Epps [a,b], Haizhou Li [c], Ting Dang [d]

[a] School of Electrical Engineering and Telecommunications, UNSW, Australia
[b] Data61, CSIRO, Australia
[c] Dept of Electrical & Computer Engineering, NUS, Singapore
[d] Dept of Computer Science and Technology, University of Cambridge, Cambridge, UK

ARTICLE INFO

Keywords:
Adaptive feature extraction
Anti-spoofing
Auditory system modeling
Deep learning
Automatic speaker verification

ABSTRACT

Developing robust countermeasures to protect automatic speaker verification systems against replay spoofing attacks is a well-recognized challenge. Current approaches to spoofing detection are generally based on a fixed front-end, typically a time-invariant filter bank, followed by a machine learning back-end. In this paper, we propose a novel approach whereby the front-end comprises an adaptive filter bank with a deep neural network-based controller, which is jointly trained along with a neural network back-end. Specifically, the deep neural network-based adaptive filter controller tunes the selectivity and sensitivity of the front-end filter bank at every frame to capture replay-related artefacts. We demonstrate the effectiveness of the proposed framework in spoofing attack detection on a synthesized dataset and ASVSpoof 2019 and ASVSpoof 2021 challenge datasets in terms of equal error rate and its ability to capture artefacts that differentiate replayed signals from genuine ones in comparison to conventional non-adaptive front-ends.

1. Introduction

Automatic speaker verification (ASV) is now being used in security-sensitive applications like mobile banking and financial services (Nuance 2023; Australian Taxation Office, 2021). Despite technological advances, the vulnerability of ASV systems to spoofing attacks (also known as presentation attacks) has become a great concern (Evans et al., 2013, Wu et al., 2015). The types of spoofing attacks that have been identified are impersonation, speech synthesis (SS), voice conversion (VC) and replay attacks (Lau et al., 2004, Lindberg and Blomberg, 1999, Wu and Li, 2013, Wu et al., 2014, Hautamäki et al., 2015; Gao et al., 2021). Speech synthesis and voice conversion generate artificial speech to mislead an ASV system, whereas a replay attack involves recording a speaker's voice and playing it back to the ASV system. Among these, replay attacks are the easiest to deploy, requiring only high-quality, low-cost consumer-level recording and playback devices such as smartphones.

The success of replay detection solutions depends on the reliable extraction of artefacts introduced during recording and playback of speech signals. The development of robust countermeasures for replay attacks is particularly challenging due to the varied nature of the quality of the attacks. If a covert recording was done using a low-quality device or within a noisy environment, the replayed signal would contain additive and convolutive noise which can be easily detected. However, a recording made using a high-quality device, within a quiet environment, may be nearly indistinguishable from the genuine signal (Kinnunen et al., 2017, Delgado et al., 2018).

Most speech processing systems employ a front-end, typically consisting of a set of parallel filters, which carries out spectral decomposition of the input speech. In end-to-end deep neural network (DNN) based systems, this front-end is generally implemented as one or more convolutional layers, whereby the filter characteristics are learnt from the training data (Dinkel et al., 2017, Sailor et al., 2018). Most well-known spoofing detection systems also follow the similar approaches (Font et al., 2017, Lavrentyeva et al., 2017, Gunendradasan et al., 2018, Zeinali et al., 2019). However, in all these cases, the filters are fixed, and their characteristics do not change when the system is in use.

Psychoacoustic research (Lyon, 1990, Elliott and Shera, 2012) has shown that the mammalian cochlea has an active mechanism that amplifies its response at low input signal levels and saturates at high input signal levels, leading to an adaptive dynamic range compression mechanism. It is known that such adaptive mechanisms have led to

* Corresponding author at: School of Electrical and Computer Engineering, Purdue University, West Lafayette IN, USA.
E-mail address: wwickram@purdue.edu (B. Wickramasinghe).

https://doi.org/10.1016/j.specom.2023.102973
Received 6 February 2022; Received in revised form 6 August 2023; Accepted 17 August 2023
Available online 20 August 2023
0167-6393/© 2023 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

remarkable selectivity and sensitivity in the cochlea. Examples of adopting this compressive non-linear tuning mechanism can be found in speech processing applications such as speech recognition (Ambikairajah and Kilmartin, 1991), speech enhancement (Darlington and Campbell, 1996, Hussain and Campbell, 1998), speaker recognition (Islam et al., 2016) and the authors' previous work on spoofing attack detection (Wickramasinghe et al., 2019a; Gunendradasan et al., 2019). These models have shown superior robustness against varied noise conditions compared to conventional models.

In our previous work, an auditory-inspired input level-dependent filter tuning mechanism, where the filter gains are selectively increased when the input energy is low, was proposed for the task of replay spoofing attack detection in ASV (Wickramasinghe et al., 2019a; Gunendradasan et al., 2019). Models for parallel (Wickramasinghe et al., 2019a) and cascaded (Gunendradasan et al., 2019) filter banks were built upon the hypothesis that previously undetectable replay artefacts with low energy could get boosted using an adaptive filter bank. Enhancing such artefacts could help differentiate a genuine signal from a replayed one. While this adaptive framework improved detection performance over a non-adaptive front-end, the adaptation process does not consider the replay attack detection performance.

It is known that replay artefacts that arise from different replay devices and acoustic environments reside in varied frequency bands and/or temporal segments of an utterance and cannot be defined using a simple function (Delgado et al., 2018). Consequently, capturing different types of artefacts calls for both subband and frame-specific attention, which typical filter banks lack. While input level-dependent adaptation spans temporal and frequency dimensions, replay artefacts are not considered during adaptation. The challenge of designing such an adaptive framework is that suitable filter tuning characteristics that capture different replay artefacts are unknown. Therefore, developing a relationship between required filter gains and replay artefacts is not straightforward.

In this paper, we leverage the learning capabilities of a deep learning model to learn the replay artefact-related information by dynamically identifying the most informative subbands for each frame of the input speech frame. To tackle the problem of having no reference tuning characteristics, we propose a DNN-based adaptive controller (depicted in red in Fig. 1) to propagate feedback from the back-end to the front-end, enabling its adaptivity based on the replay attack detection performance. The adaptive controller is jointly optimized with a replay attack detection back-end to identify signal components that contain replay artefacts and locally enhance them by increasing the filter selectivity and gain. The proposed framework integrates knowledge-based signal processing concepts of the human auditory system with the data-driven learning capabilities of neural networks.

2. Related Work

2.1. Adaptive DNN framework

Some examples of DNN-based adaptive filtering frameworks can be found in the literature. A convolutional neural network (CNN) model which approximates the cochlea tuning characteristics including level-dependent adaptation has been proposed in (Baby et al., 2021). The model has been trained on simulated cochlea tuning curves generated using a state-of-the-art analytical cochlea model. But, as discussed in Section 1, adopting a similar framework for replay attack detection is challenging since desired filter tuning characteristics are unknown. Adaptive frameworks have also been utilized in automatic speech recognition to adapt models to speakers or acoustic environments. For example, Delcroix et al. proposed an auxiliary adaptive network to learn the context weights corresponding to a class of speakers (Delcroix et al., 2018). Wan et al. have presented an attention-based mechanism for speaker adaptive training via speaker embeddings (Wan et al., 2020). In a recent work by Zhang et al., an adaptive attention mechanism has been proposed for speech enhancement (Zhang et al., 2021). Here, a DNN-based module inspired by Squeeze and Excitation attention (Hu et al., 2018) has been used to dynamically assign weights to frequency elements of a temporal convolutional network. The framework involves a global averaging pooling layer, which averages out the temporal information of the input feature prior to weight calculation. These frameworks only focus on frequency band adaptation since context information for speaker adaptation is generally constant over an utterance. It is hypothesised that both frequency and temporal artefacts are crucial for replay attacks and should be learned adaptively.

An adaptive CNN which reflects phone-variation of a speech signal has been proposed for speaker recognition in (Kim and Park, 2021). Here, frame-level features are extracted via input frame-dependent kernels calculated using a specifically designed CNN. The importance of adaptive feature extraction to identify most informative subbands or segments has recently been studied in artificial spoofed speech detection as well (Tak et al., 2021, Li et al., 2021). While time-dependent adaptive front-ends are just gaining traction in speech processing area, such systems can be found more commonly in image processing systems. Current work takes inspiration from these approaches. De Brabandere et al. have proposed a network where filters are dynamically generated conditioned on an input, for video and stereo prediction (De Brabandere et al., 2016), where the adaptive filters were shown to capture the motion patterns of different videos better. The framework consists of a filter generating network which was optimized during training for video and stereo prediction. Networks of this form have also been used in other areas such as short-range weather prediction (Klein et al., 2015) and in recovering high-resolution images from low-resolution ones (Riegler et al., 2015). Kang et al. have presented an adaptive CNN where convolutional filter weights adapt to the input scene context via auxiliary information (Kang et al., 2017). Here, filter weights are learnt using a low-dimensional 'filter manifold' network which takes side information about the context as input.

However, these adaptive systems either rely on the input alone or employ auxiliary networks to generate convolutional filter kernels which can be difficult to interpret. We employ filters with known responses which are adapted using a single parameter through a learnt adaptive controller. In this paper, we propose a replay attack detector adaptively tuned on both the input characteristics and the replay-related information from the back-end obtained via the feedback path.

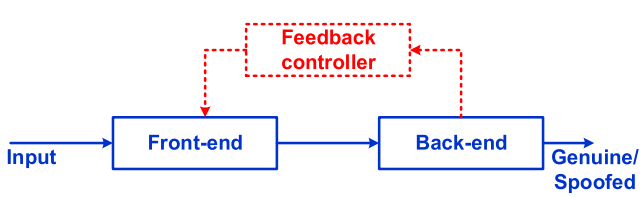

Fig. 1. Proposed adaptive controller-based framework.

2.2. Replay spoofing attack detection

Many hand-crafted features for replay attack detection have been proposed in recent years. Traditional spectral decomposition using a fixed filter bank has been the most popular front-end for feature extraction. Conventional cepstral analysis is one of them (Font et al., 2017, Witkowski et al., 2017). However, most successful systems have taken rather different approaches from cepstral analysis. For example, a widely accepted baseline for spoofing detection is Constant Q Cepstral Coefficients (CQCC) (Delgado et al., 2018, Todisco et al., 2017). It uses constant Q transform (CQT) instead of short-term Fourier transform (STFT) to obtain time-frequency representations. CQT employs geometrically spaced frequency bins as opposed to uniformly spaced bins as in STFT.

Apart from the hand-crafted features, representations learnt from

deep learning frameworks have also led to competitive results. Examples include features extracted from a Light Convolutional Neural Network (LCNN) with log spectrogram inputs (Lavrentyeva et al., 2017), Siamese deep embeddings extracted from spectrograms using hybrid CNN-based frameworks (Sriskandaraja et al., 2018), utterance level embeddings extracted using a gated recurrent CNN (Gomez-Alanis et al., 2019) with spectrograms and group delay inputs and attention weighted features extracted from an attention based Residual Network (ResNet) with group delay inputs (Tom et al., 2018). Sailor et al. have explored the use of Convolutional Restricted Boltzmann Machine to learn a filter bank using genuine and replayed data (Sailor et al., 2018). SincNet, a CNN where the first layer comprises parameterized sinc functions (Ravanelli and Bengio, 2018a; Ravanelli and Bengio, 2018b), has been recently proposed as a means to learn filters with a clear physical meaning unlike typical convolutional layers. The use of SincNet in the context of spoofing attack detection has been studied in (Zeinali et al., 2019). However, the work does not provide an in-depth analysis of the learnt filters. All these approaches focus on identifying a fixed filter representation with the ability to capture varied replay characteristics.

Hence, despite these advances, the challenge of developing systems that can generalize to unseen replay conditions and model a wide range of replay characteristics persists. To address this challenge, we focus on a DNN-based framework, which dynamically identifies the most informative subbands using feedback from a back-end classifier and dynamically enhances signal components that differentiate genuine speech from replayed speech.

3. Proposed framework

3.1. Overview

Our previous work (Wickramasinghe et al., 2019a) proposed a feed-forward adaptive front-end to adjust filter characteristics based on input energy levels in a pre-determined manner. In this paper, we propose the addition of a novel neural network-based feedback controller to the framework. In addition, we also present the joint training methodology necessary to train this feedback controller using a DNN-based back-end. The proposed DNN-controlled adaptive front-end is depicted in Fig. 2. Unlike conventional systems, it includes a DNN subsystem that dynamically generates filter coefficients for the front-end filter bank, allowing it to adapt to input signal characteristics. Specifically, the front-end consists of two finite impulse response (FIR) filter banks: the first set of filters with fixed unit gain (Layer 1 in Fig. 2) maintains narrow bandpass filtering, while the second set (Layer 2 in Fig. 2) comprises of adaptive filters, where the selectivity of each filter is controlled using its quality-factor (Q-factor, $Q\{x\}$ in Fig. 2).

The adaptive Q values of the second set of filters is determined by both the input signal (denoted by x) power within that subband and feedback from the back-end. Two components of the Q-factor, $Q^E\{E\{x\}\}$ and $Q_\phi^{FM}\{G\{x\}\}$, are calculated for each filter in the second filter bank via two distinct paths (shown in green and red in Fig. 2) and an updated set of filter coefficients are computed based on their sum, $Q\{x\} = Q^E\{E\{x\}\} + Q_\phi^{FM}\{G\{x\}\}$. The first component, $Q^E\{E\{x\}\}$, is computed from the energy $E\{x\}$ of spatially differentiated filter output of the first filter bank, while $Q_\phi^{FM}\{G\{x\}\}$ is inferred by a DNN-based adaptive controller that takes frequency modulation (FM) components $G\{x\}$ of the second filter bank output as input. Here, the superscripts E and FM have been used to differentiate energy-based and FM-based Q-factors, respectively, and ϕ denotes the tunable weights of the adaptive controller. Section 3.4.1 discusses $G\{x\}$ in detail.

By constraining the adaptive filters to be FIR filters with a known response and only the Q-factor as an adjustable parameter, we achieve three benefits: (a) the number of learnable parameters is reduced to the Q-factor instead of the entire filter response, analogous to the approach adopted in (Ravanelli and Bengio, 2018a), b) reduces the chances of overfitting and vanishing gradients (Seki et al., 2019); and (c) the filters can be placed on an appropriate frequency scale (for e.g., equally spaced on the Bark scale). All components of the proposed framework are differentiable, allowing us to employ gradient-based optimization to update the parameters of the DNN-based adaptive controller jointly with the parameters of the back-end classifier. It should be noted that back-propagation is not straightforward since the direction of information flow through the adaptive controller is from a later stage of the system to an earlier one. The paths for gradient flow to train both the back-end and the controller are shown as Backprop1 and Backprop2 in Fig. 2. This joint training also enables the controller to be trained without any labelled Q-factors, by optimizing for a reduction in spoofing detection error.

Once trained, the adaptive controller is expected to adapt dynamically to better capture signal components that differentiate replayed signals from genuine ones. This Q control must be adaptive since the components that need to be enhanced, and the factors that affect this are transient, i.e., certain bands need to be enhanced at certain times. Detailed system configurations are given in the following sections.

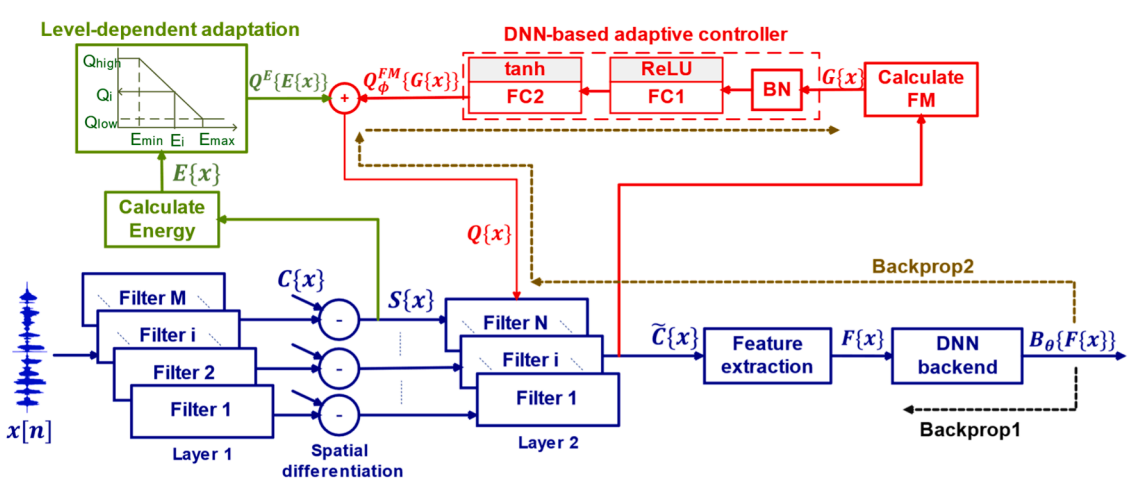

Fig. 2. The DNN based feedback controller and the spoofing detection back-end are jointly trained with gradients computed along two different paths. The frequency modulation (FM) components ($G\{x\}$) calculated per frame from Layer 2 output ($\tilde{C}\{x\}$) are used as input to the adaptive controller. In this figure the path along which gradients of the back-end are estimated is depicted as Backprop1 and the path along which the gradients of the controller are estimated is depicted as Backprop2. The filter selectivity varies as Q-factor varies (in the Layer 2 filters). Increased Q-factor increases filter gain, while reducing the bandwidth (i.e., higher selectivity). In the Figure, FC1 and FC2 are used to denote fully connected layers and BN is used to denote batch normalization.

3.2. Subband decomposition

The first step in the proposed framework, as shown in Fig. 2, is subband decomposition where a 1-dimensional time convolutional layer acting as a filter bank was employed (shown in blue in Fig. 2). The Layer 1 consists of M narrowband CNN filters, $C = \{c_1, c_2, \ldots, c_M\}$; and the Layer 2 consists of N tunable CNN filters, $\widetilde{C} = \{\widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_N\}$ and k^{th} order spatial differentiation (further details in Section 3.3) is carried out between the two layers. Here, c_i and $\widetilde{c}_i$ represent the ith filter in Layer 1 and Layer 2, respectively. M and N are the number of filters in each layer such that $N = M - k$. It should be noted that each channel output from Layer 1 is filtered through the corresponding channel in Layer 2. Center frequencies of Layer 2 filters were computed to match the center frequency shift that occurs due to spatial differentiation (Wickramasinghe et al., 2019b).

The coefficients (or weights) of a typical convolutional layer are learnt from the data. But, in our case, the filter coefficients were defined using known impulse responses, with their Q-factors as adjustable parameters. During adaptation, coefficients of Layer 1 were kept fixed, while Layer 2 coefficients were made adaptable. Truncated digital Gabor filter impulse responses were used to simulate the filter bank model. The primary reason for choosing Gabor filters is that it is straightforward to obtain a relationship between its Q-factor and filter gain. Also, Gabor filters show characteristics similar to those of auditory filters. The impulse response and frequency response of the chosen Gabor filter are given in Eqs. (1) and (2) respectively (Maragos et al., 1991):

$$h[n] = e^{\left(-(bn)^2\right)} \cos(\Omega_c n) \tag{1}$$

$$H(\Omega) = \frac{\sqrt{\pi}}{2b}\left(e^{\left(-\frac{(\Omega-\Omega_c)^2}{4b^2}\right)} + e^{\left(-\frac{(\Omega+\Omega_c)^2}{4b^2}\right)}\right), \tag{2}$$

where f_c is the resonant frequency, f_s is the sampling frequency, $\Omega_c = \frac{2\pi f_c}{f_s}$, $b = \sqrt{2\pi}\ \frac{BW}{2f_s}$ and the 3 dB bandwidth BW of the filter is given by $BW = \frac{f_c}{Q}$. The relationship between the gain at the resonant frequency and the Q-factor of the Gabor filter can be obtained by evaluating (2) at $\Omega = \Omega_c$, which results in $|H(\Omega_c)| = \frac{\sqrt{2}\pi.Q}{\Omega_c}(1 + e^{-8\pi Q^2})$. It can be shown that for Q-factors larger than 0.5 ($Q \geq 0.5$), the factor $e^{-8\pi Q^2}$ is very small and therefore,

$$|H(\Omega_c)| \approx \frac{\sqrt{2}\pi.Q}{\Omega_c} \tag{3}$$

3.3. Level-dependent adaptation

In our previous work (Wickramasinghe et al., 2019a), we proposed a classic machine learning-based feed-forward adaptive front-end (without deep learning), which consists of two filter banks, providing variable selective gain via Q-factors based on the input energy levels. The Q-factor ($Q\{x\} = Q^E\{E\{x\}\}$) is high for an input with a small amplitude, leading to a filter with increased gain while reducing the bandwidth, making the filter more selective and amplifying the input signal. Contrastively, the Q-factor is low for input with a large amplitude, leading to a filter with decreased gain while increasing the bandwidth, making the filter less selective and compressing the input signal. Differing from our previous work (Wickramasinghe et al., 2019a), the current framework replaces the filter banks with a 1-dimensional convolutional layer to allow for gradient propagation and employs a DNN-based adaptive controller to modify the Q-factor based on the back-end output.

The input signal $x[n]$ was filtered frame-wise using the Layer 1 filter bank and the energy corresponding to each band was calculated over the spatially differentiated filter output. For uniformly spaced filters, first-order spatial differentiation is implemented by taking the difference between outputs of two adjacent filters and subsequent higher-order spatial differentiations can be implemented by repeating this process. It has been shown that the spatial differentiation increases the sharpness of filter transition bands, reducing their overlap, which in turn reduces the correlation between adjacent feature coefficients (Wickramasinghe et al., 2019b). In our previous work, replay attack detection using this level-dependent adaptive front-end was shown to reduce error rates compared to conventional non-adaptive front-ends (Wickramasinghe et al., 2019a).

3.4. DNN-based adaptive controller

3.4.1. Configuration

The adaptive controller (depicted in red in Fig. 2) takes context information from within the model as input ($G\{x\}$) to determine $Q_{\phi}^{FM}\{G\{x\}\}$. As shown in Fig. 2, the frame-averaged frequency modulation (FM) components (Alsteris and Paliwal, 2007, Liu et al., 1997) of the subband signals were used as the input to the controller network. There is evidence in psychoacoustic research to show that the FM component is complementary to the amplitude modulation (AM) component within the auditory system and captures spectral information for accurate signal-noise separation for speech perception (Nie et al., 2004). Additionally, phase information carried in the FM component reflect reverberation effects and spectral envelope changes due to channel characteristics (Belhomme et al., 2016, Patil et al., 2017). In this work, Q-factors that depend on FM components are combined with Q-factors depending on input energy (relating to AM) via addition operation.

The FM component ($G\{x\}$) was calculated from the Layer 2 output and given as input to the adaptive controller. Spectral centroid deviation method proposed in our previous work (Gunendradasan et al., 2018) was used for FM calculation. For a subband i with center frequency f_{ci}, the FM component FM_i is given by:

$$FM_i = |SCF_i - f_{ci}| \tag{4}$$

$$SCF_i = \frac{\sum_{r=r_l}^{r_u} r|S[r]|}{\sum_{r=r_l}^{r_u} |S[r]|} \tag{5}$$

where SCF_i is the spectral centroid frequency (SCF) of the subband, r_l and r_u are the lowest and the highest frequency limits of subband i respectively, and $S[r]$ is the spectral magnitude at frequency r. SCF is a measure of the 'center of gravity' of the subband. Hence, SCF is dependent on the bandwidth of the filter used to create the subband. Therefore, the FM component of a given subband reflects the changes in selectivity within that subband. The DNN based adaptive controller architecture consists of a batch normalization (BN) layer to scale down the FM input and a fully connected layer ($FC1$) with rectified linear unit (ReLU) activation ($max\{0, BN(G\{x\})\}$). Hence, during training, the layer either calculates an additional factor added to the level-dependent Q-factor ($Q^E\{E\{x\}\}$) or keeps it zero depending on the importance of the subband component.

It is important to note that increasing of Q-factors of filters increases the filter bank ripple. Therefore, care should be taken to ensure that the filter bank ripple with respect to total magnitude spectrum is not too high as it may degrade the detection performance (Wickramasinghe et al., 2019a, Kua et al., 2010). To limit the maximum Q-factors generated by the $FC1$ layer in a differentiable manner, a scaled and shifted version of a fully connected layer ($FC2$) having diagonal weight matrix w and biases b followed by a tanh non-linearity was used. As shown in Fig. 3, for each filter i, $FC2$ layer maps the Q-factor Q_i^{In} generated by $FC1$ layer into $0 \leq Q_i^{Out} \leq Q_i^{max}$ where Q_i^{max} is the maximum Q-factor that filter i can take. Parameters of $FC2$ layer and constants $\{\alpha_1, \alpha_2, \ldots, \alpha_N\}$ and $\{\delta_1, \delta_2, \ldots, \delta_N\}$ were predetermined. Q_i^{max} for each filter was determined to match the maximum suitable impulse response length of the filter. Maximum impulse response

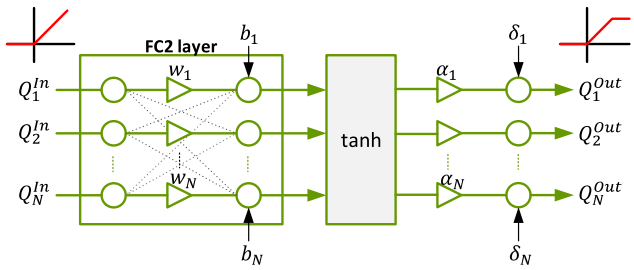

Fig. 3. Q factor saturation to set upper limits using a fully connected layer followed by tanh activation in the output of FC2 of the adaptive feedback controller shown in Fig. 2.

lengths were fixed to the input signal frame length to avoid inaccuracies of filter outputs.

3.4.2. Training

Features, $F\{x\}$, were extracted from the output of the Layer 2 filters and given as input to a DNN back-end, $B_\theta\{\cdot\}$, with parameters θ. Backprop1 in Fig. 2 shows the typical backpropagation path of a DNN-based spoofing detection system where feature extraction is followed by a DNN back-end. Here, the optimization problem takes the following form:

$$\widehat{\theta} = \underset{\theta}{argmin}\mathscr{L}(y, B_\theta\{F\{x\}\}) \tag{6}$$

where, $\mathscr{L}(\cdot,\cdot)$ is a loss function measured between the target label y and predicted output $B_\theta\{F\{x\}\}$ for a given utterance x.

The backpropagation path of the proposed adaptive framework takes a different form and is given by Backprop2 (See Eq. (7)). Layer 2 filter coefficients now depend on Layer 1 filter output energies, transformed level dependent adaptation block, as well as the adaptive controller output ($Q_\phi^{FM}\{G\{x\}\}$); ϕ denotes the tunable weights of the adaptive controller, specifically the weights of the $FC1$ layer. The new optimization problem is given below.

$$(\widehat{\phi}, \widehat{\theta}) = \underset{\phi,\theta}{argmin}\mathscr{L}\left(y, B_\theta\left\{\widetilde{C}\left\{x, \widetilde{Q}_\phi\{G\{x\}\}\right\}\right\}\right) \tag{7}$$

The adaptive controller uses the current frame's information to process Q-factors for the next frame, behaving like a recurrent neural network. Frame-wise processing of the whole utterance is followed by feature matrix calculation and DNN-based classification. Since Q-factor calculation is recursive, the controller network is unrolled through time during backpropagation. That is, at time T, the derivative of the loss $\mathscr{L}$ with respect to ϕ is given by:

$$\frac{\partial \mathscr{L}^T}{\partial \phi} = \sum_{t=1}^{T} \frac{\partial \mathscr{L}^T}{\partial \phi_t} \tag{8}$$

Each input frame represents a time step t.

3.5. Feature extraction

Although features can be learnt within the network itself, constraining the feature extraction to a deterministic process reduces the number of parameters within the network. Due to the recurrent nature of the adaptive controller, gradient vanishing issues could occur. Therefore, reducing the number of learnable parameters is important in addition to avoiding overfitting. Two features were considered for evaluation of the model: subband energy and spectral envelope centroid magnitude.

Subband energy is the typical energy component extracted from a subband signal. It can be defined as the mean magnitude of the Fourier transform of a subband signal. In our previous work, the spectral envelope centroid magnitude (CM) feature was proposed as a feature for spoofing detection (Wickramasinghe et al., 2019b). It is defined as the weighted average magnitude of the spectral envelope of a particular subband signal. The spectral envelope is extracted by taking the Fourier transform of a full-wave rectified subband signal. In our previous work, CM was extracted from the region up to 950 Hz. But in this work, the full band subband spectral envelope (0–8 kHz) was divided into 5 octaves and CM was extracted from each octave. CM of pth octave of the subband signal of a selected frame is given by:

$$CM_p = \frac{\sum_{f=f_l}^{f_u} f.|W[f]|}{\sum_{f=f_l}^{f_u} f} \tag{9}$$

where f is the frequency of each component, f_l and f_u are the lower and upper-frequency limits of the octave component and $|W[f]|$ is the spectral envelope.

4. Experimental setup

4.1. Database

The ASVspoof 2019 Physical Attack (PA) database was primarily used for the experiments in this work (Wang et al., 2020). This database consists of genuine speech signals and simulated replayed signals which emulate the effect of recording and playback devices and reverberant acoustic environment (Wang et al., 2020). Signals have been sampled at 16 kHz. Training and development data have been created according to 27 different acoustic and 9 different replay configurations. Replay configurations have been defined based on the combination of attacker-to-speaker recording distance and loudspeaker quality. All three partitions, training, development, and evaluation are disjoint from each other. Having simulated samples, this database presents a controlled evaluation setup for the study of replay attack detection.

The proposed system trained on the ASVspoof 2019 database was also evaluated on ASVspoof 2021 database (Yamagishi et al., 2021). This recently released database consists of genuine speech signals and real replayed signals generated using different physical devices within varied acoustic conditions.

4.2. Performance evaluation measures

Two performance evaluation measures, equal error rate (EER) and tandem detection cost function (t-DCF) were used in the experiments. Since EER is used in most spoofing detection studies, it was used as the primary evaluation metric in the experiments. Minimum normalized t-DCF is a recently introduced evaluation measure to evaluate the combined performance of ASV and spoofing countermeasures (Wang et al., 2020).

4.3. System parameters

Speech signals were windowed into non-overlapping frames of 11 ms because a small frame size was necessary to emulate near real-time adaptation. However, the frame size should be long enough to accommodate the filter impulse response length for the filtered outputs to be accurate. Frame overlapping was not required due to the short frame size. Based on the mean utterance length of the database, all feature matrices were made the same size by retaining the first 400 frames from each utterance, and in cases where utterances were shorter, the utterance was repeated to make 400 frames. Empirical results showed that frame counts higher than 400 do not significantly improve performance, while those lower than 400 degrade the performance. 40 normalized Gabor filters were placed uniformly in Hz frequency scale in Layer 1. Since spatial differentiation sharpens the filters, the filtered signals were subjected to one spatial differentiation. The resulting 39 filtered outputs per framed signal were then filtered through adaptive filters in Layer 2. As highlighted in our previous work (Wickramasinghe et al., 2019b), spatial differentiation shifts the subband center frequencies to the mean

values of the center frequencies used in spatial differentiation. Hence, the center frequencies of Layer 2 were calculated accordingly. Additionally, an appropriate number of samples from the previous frame were carried into the current frame to avoid distortions introduced by frame boundary issues during frame-wise filtering.

Once a batch of utterances was filtered, concatenated subband signals were transformed into features. Subband energy and CM features were extracted from 22 ms frames with 11 ms overlap. Although 11 ms non-overlapping frames were used during filtering, preliminary experiments showed that frames longer than 11 ms with overlap were required for better detection by the back-end. The reason could be that longer temporal patterns probably convey more useful information regarding genuine and replayed signals. Log and discrete cosine transform (DCT) were applied to decorrelate the features and the first 20 coefficients were retained. Once features were extracted, their delta and delta-delta coefficients were appended to include additional temporal information (referred to as $F(x)$ in Fig. 2).

The model was trained by minimizing the cross-entropy loss function. Adam optimizer with a learning rate of 0.001 was used for optimization with a batch size of 48. The model was trained for 35 epochs. PyTorch Toolkit[1] was used to build and train all the models (Paszke et al., 2019). Parts of code by Ravanelli et al. were used in the implementation (Ravanelli and Bengio, 2018b). A summary of the system parameters is given in Appendix A.

4.4. Back-end

A neural network-based back-end classifier was required for the system to be trained in an end-to-end manner. A deep residual convolutional network inspired by the work of Alzantot *et al.* was used as the back-end (Alzantot et al., 2019). One reason for the choice is residual architectures have given encouraging results for replay attack detection problems. While the basic architecture was similar to (Alzantot et al., 2019), some hyperparameters were adjusted based on the feature dimensions and development set performance. The architecture starts with a 2-dimensional convolutional layer (conv2D). This is followed by 5 residual blocks (as opposed to 6 residual blocks in (Alzantot et al., 2019). Outputs of the residual blocks are fed into a dropout layer followed by a fully connected layer with leaky-ReLU activation and finally fed into the last fully connected layer that generates posterior probabilities of the two classes using a log softmax activation.

The input feature matrix (subband energy or CM feature) is first passed through the conv2D layer with filter size $= 3 \times 3$ and stride $= 2$. The output consists of 32 feature maps. Next, the feature maps are sent through the residual blocks and the output dimension is then reduced using a max-pooling layer. Then a dropout layer is followed by a fully connected layer (referred to as *FC1-back-end*) with 128 output neurons and leaky ReLU activation and the final fully connected layer which produces the predictions through a log softmax activation. The residual block is shown in Fig. 4. The block structure is similar to that in (Alzantot et al., 2019), where each residual block consists of a conv2D layer with 32 filters, followed by batch normalization, leaky ReLU activation and dropout and finally another conv2D layer. Based on the input size, stride lengths of the conv2D layers are set to 2 (as opposed to stride=3 in (Alzantot et al., 2019). Skip connections add the input to the output via a conv2D layer used to match the dimensions.

5. Evaluation

The proposed adaptive controller-based framework was trained using the training set of the ASVspoof 2019 PA database and evaluated on three sets of data. One set consists of a subset of genuine data from the database and replayed signals synthesized by the authors by adding artefacts to the genuine data. The second and third sets are publicly released evaluation sets of the ASVspoof 2019 database and the ASVspoof 2021 database.

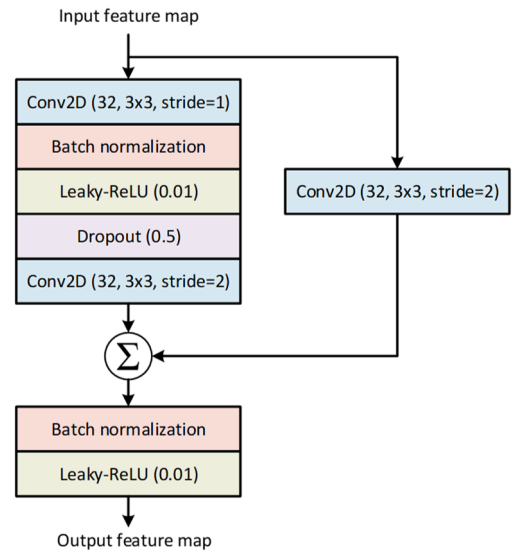

Fig. 4. Structure of the residual block used in the back-end classifier.

5.1. Framework analysis using synthesized data

The proposed DNN-based adaptive framework was built on the hypothesis that the adaptive controller captures artefacts that differentiate genuine and replayed signals and enhances those artefacts using filter adaptation. Hence this model should increase the separation between genuine and spoofed data compared to a non-adaptive model or a level-dependent adaptive model. To evaluate this, a set of genuine data was chosen and transformed into spoofed signals. Synthesized data ensured that the replay artefacts were the only difference between the genuine data and their corresponding spoofed signals. Additionally, having spoofed signals with known replay artefacts facilitated a better analysis of the framework. Evaluations were carried out based on the following factors: 1) generated Q-factors, 2) extracted features and 3) classification results.

The model was first trained using the training set of the ASVspoof 2019 database. Then the trained model was analyzed using a synthesized set of data. 1000 genuine signals from the same training set were used as the genuine set of the synthesized set. Spoofed signals were generated by adding reverberation effects and noise into the genuine signals. Reverberated signals were created by convolving genuine signals with room impulse responses generated using Roomsimove toolkit (Vincent, 2016) and then adding band-limited white noise at different time instances to simulate additional replay artefacts. The room impulse response simulates the reverberation of an empty room with each of the six surfaces having a different absorption factor with the maximum reverberation time (RT60) being 0.71 s. Additive noise centered around 2 kHz, 4 kHz and 6 kHz and having signal-to-noise-ratios 30 and 50 were added to randomly chosen time segments. Fig. 5(a) and (b) show spectrograms of a sample genuine signal and its spoofed signal respectively. Fig. 5 (b) also indicates the artefacts resulted from added noise and reverberation. Temporal smearing effects caused by reverberation can be seen throughout frames across most frequency bands.

5.1.1. Q-factor analysis

Firstly, it was verified that the Q-factors generated by the proposed framework correlate to the replay artefacts. When effectively visualizing this correlation, effects of other variabilities such as speech content and speaker characteristics should be minimized and replay artefacts should

[1] Our code is available at: https://github.com/buddhiW/AdaptiveNet.

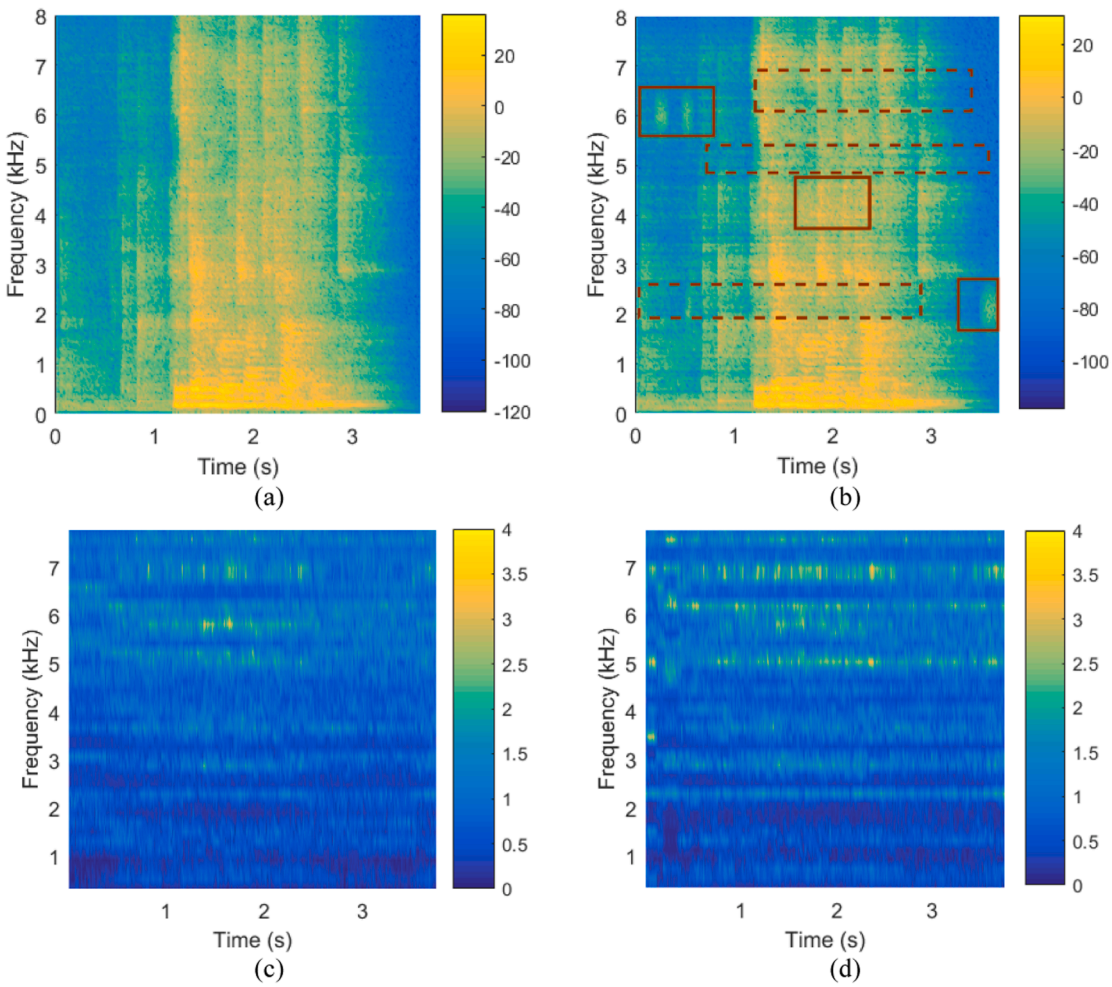

Fig. 5. (a) and (b) show genuine and spoofed spectrogram samples from the synthesized dataset, respectively. Added noise is shown in solid lines and some of the reverberation-related temporal smearing is shown in dashed lines. The average variation of Q-factors per frame per subband generated by the framework is shown for genuine (c) and spoofed (d) signals.

be enhanced. Thus, Q-factors calculated by the trained network were averaged across speech signals for genuine and spoofed cases. Q-factor variations over time and frequency for genuine and spoofed signals are shown in Fig. 5(c) and (d). When comparing the four figures in Fig. 5, the Q-factors resulted from spoofed signals are generally higher than those from genuine signals. This variation can be associated with the regions affected by reverberation. Also, the effect of the noise added around 6 kHz band is reflected in the Q-factors up to around 300 ms. It is interesting to note that Q-factors across the whole frequency band are affected by the noise. In general, it can be concluded that the DNN of the adaptive framework has captured replay related information as reflected by the Q-factor variation.

5.1.2. Feature space visualization

Simulated signals were used to compare five models trained on the training set of ASVspoof 2019 database: (a) proposed adaptive controller-based framework, referred to as *Energy+FM* model and explained in Section 3.4 (Red + Blue + Green in Fig. 2), b) level-dependent adaptive framework, referred to as *Energy only* model and discussed in Section 3.3 (Green + Blue Fig. 2), c) non-adaptive framework, referred to as *Low-Q* model, with Layer 2 filters set to unit gain (Blue in Fig. 2), d) non-adaptive framework, referred to as *High-Q model,* with Layer 2 filters set to the upper limit Q^{max} of each filter (Blue in Fig. 2) and (e) a CQCC-based model (Todisco et al., 2016). CQCC features maintain constant Q-factors across the frequency spectrum and are a standard baseline in spoofing literature. In systems (a)-(d), DNN back-ends were employed to train models that utilize CM features for differentiating between genuine and spoofed input signals. For system (e), the same back-end was utilized to classify CQCC features. Development set of the ASVspoof 2019 database was used for model validation during training.

During the evaluation, the back-end network was tapped into at the *FC1-back-end* layer and 128-dimensional neural network features were extracted using all five frameworks. The feature spaces of both genuine and spoofed signals were visualized by projecting them into 2-dimensional space using t-SNE toolkit (Van der Maaten and Hinton, 2008). t-SNE is an unsupervised technique proposed to visualize high-dimensional data using a 2-dimensional map, while retaining most of the original information. It maps each high-dimensional data point to a location in a 2-dimensional map, where similar points are modelled by nearby points and dissimilar points are modelled by distant points. Fig. 6 shows the visualisations.

The figure conveys that, features extracted using the proposed framework (Fig. 6(a)) show lower intra-cluster variance within both classes compared to the other four models. In comparison to the *Energy only* model, we can see that the *Energy+FM* model has captured more class-specific components. Also, compared to *Low-Q, High-Q* and *Energy only* methods, a more significant cluster separation between genuine and spoofed features is observed. Since the only difference between the genuine and spoofed signals are the replay artefacts added to the genuine signals, it can be assumed that the proposed framework has been able to capture those artefacts, hence resulting in better clustering.

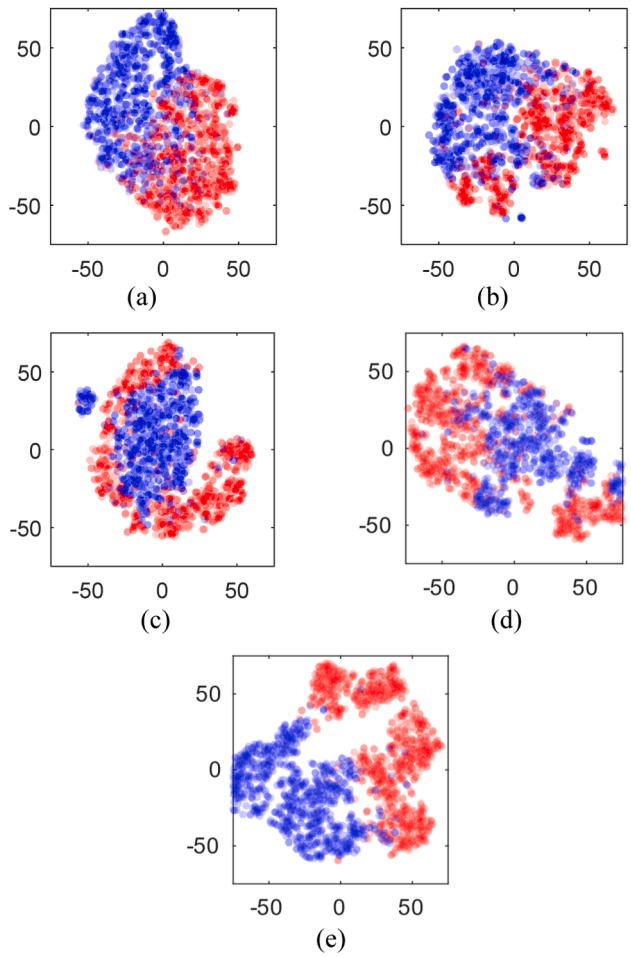

Fig. 6. t-SNE-based 2-dimensional space using features extracted from (a) proposed DNN framework (*Energy*+*FM*), b) level-dependent filter framework (*Energy only*), c) non-adaptive filter framework (*Low-Q*), d) non-adaptive filter framework (*High-Q*), and (e) CQCC-based framework. Genuine features are shown in red and spoofed features are shown in blue.

While CQCC features demonstrate good class separation, Section 5.2 reveals that the proposed features outperform CQCC features when detecting high-quality replay signals.

5.1.3. Modeling and classification performance

In addition to the visual comparison, features were also mapped into a probability space to quantify the discriminating ability of the filtering methods. For this purpose, gaussian mixture models (GMMs) were used. Since the purpose of this set of experiments was to evaluate the proposed model's ability to increase the separation between genuine and replayed data, a generative classifier such as GMM was deemed suitable. Additionally, the synthesized dataset is small in size, and thus is not suitable for evaluation using a DNN-based classifier. Genuine and spoofed CM features extracted using the framework and CQCC features were modelled using 2 GMMs with 8 mixture components in each and their ability to classify unseen utterances was evaluated in terms of log-likelihood ratio. The number of GMM components was chosen based on the dataset size and feature dimensions. Here, GMMs were generated using a different set of genuine signals (different speakers and speech content) and their spoofed counterparts and previously described 2000 samples were classified using the models. Training spoofed data contained noise at different time segments from the evaluation spoofed data. The classification performance of the frameworks in terms of EER are given in Table 1.

Classification results show that the proposed framework has outperformed the *Low-Q* and *Energy only* frameworks in terms of EER. This behavior further supports the claim that the DNN-based adaptive controller has been able to enhance artefacts that separate genuine and replayed signals. While the *High-Q* and CQCC features show lower EER in this case, experiments on larger datasets show otherwise (see Sections 5.2 and 5.3).

Table 1
GMM-based classification of synthesized data using the models trained on ASVspoof 2019 dataset.

System	EER (%)
CQCC	0.18
Low-Q (Non-adaptive)	1.16
High-Q (Non-adaptive)	0.23
Energy only (Adaptive)	0.54
Energy + FM (Adaptive)	**0.30**

5.2. Framework analysis on ASVspoof 2019 database

To evaluate the effect of the adaptive filter controller on a large set of unseen data, spoofing detection experiments were conducted using the ASVspoof 2019 development and evaluation sets. The framework was evaluated using the *Low-Q, High-Q, Energy only* and *Energy*+*FM* models. In the *Low-Q* model, the Q-factors of Layer 2 were fixed at selected low-Q values and the *High-Q* model has Q-factors fixed at selected high-Q values. Fig. 7 depicts the range of Q-factors the trained *Energy*+*FM* model generated over all the time frames of the training set of the database. The low-Q and high-Q limits were fixed to the minimum and maximum Q-factors produced by the *Energy*+*FM* model (shown by black diamond shapes in Fig. 7). Subband energy and CM were used as the features and EER and t-DCF values were calculated using the probabilities output by the ResNet back-end discussed in Section 4.4. Initial experiments were done using the development set to tune the model parameters. Results obtained are given in Table 2.

Table 2 indicates that the CM feature yields the lowest EER for the *Energy*+*FM* system, whereas subband energy performs best in terms of t-DCF for the *Energy only* system. When considering the overall performance, the CM feature outperforms subband energy. Therefore, the CM feature was used for all subsequent experiments.

Table 3 presents the results of the system using the CM feature for the evaluation set along with the development set results given in Table 2. For comparison, two baseline systems are also given. *CQCC – ResNet* presents the results from a previously published work, where CQCC features are classified using a ResNet comparable to the one used in this work (Alzantot et al., 2019). *CQCC – GMM* is a system released along with the database where front-end CQCC features are classified using a

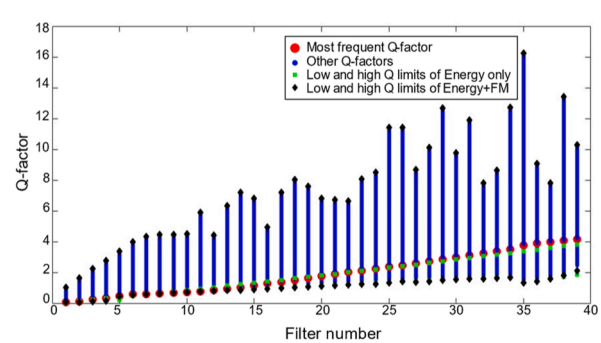

Fig. 7. Distribution of Q-factors generated by the *Energy* +*FM* system for each filter for training data. A selected vertical line represents all the Q-factors the corresponding filter is assigned throughout the time steps. The most frequently seen Q-factor is given in red. For reference, maximum and minimum Q limits of the *Energy only* framework are also given. Low-Q limit of this overlaps with that of *Energy*+*FM* model.

Table 2
Comparison of the two proposed features as measured on the development set of ASVspoof 2019.

System	Subband energy		CM	
	EER(%)	t-DCF	EER(%)	t-DCF
Low-Q	9.52	0.2572	4.54	0.1891
High-Q	4.50	0.1008	4.18	0.1065
Energy only	4.04	**0.0929**	4.44	0.0994
Energy + FM	4.22	0.1017	**3.93**	0.0934

Table 3
Experimental results (EER(%)) as measured on the development and evaluation sets of the ASVspoof 2019 database.

System	Development set		Evaluation set	
	EER	t-DCF	EER	t-DCF
CQCC – ResNet (Alzantot et al., 2019)	4.30	0.1026	**4.43**	**0.1070**
CQCC - GMM (Wang et al., 2020)	9.87	0.1953	11.04	0.2454
Low-Q - ResNet	4.54	0.1891	11.02	0.3057
High-Q - ResNet	4.18	0.1065	6.40	0.1607
Energy only - ResNet	4.44	0.0994	6.42	0.1575
Energy + FM - ResNet	**3.93**	**0.0934**	6.20	0.1480

GMM back-end (Wang et al., 2020). CQCC feature is a standard baseline used in the field of spoofed speech detection and it is based on the constant-Q transform which ensures constant Q across the entire frequency spectrum, whereas the proposed framework argues the importance of input dependent, time-varying Q-factors. The baselines considered here use variable window lengths (to maintain constant-Q) with a time shift of 8 ms.

Multiple observations can be made from the results. When compared to the selected baselines, all the systems outperform the GMM-based system,while they fail to outperform *CQCC-ResNet* system for the evaluation set. This suggests further improvements to the back-end may be required. But, since the main purpose of this work is to highlight the importance of time-varying adaptive filtering, more attention has been given to the relative performances of the four systems implemented in this work.

The proposed adaptive framework (*Energy+FM*) outperforms non-adaptive models and energy-based adaptive models in terms of both EER and t-DCF. A higher error rate for the *Low-Q* system indicates the importance of narrower filters. The *High-Q* system performs significantly better than the *Low-Q* system, especially for the evaluation set. The *High-Q* system has outperformed the *Energy only* system, which suggests that high-Q limits set for the Energy only system may not be ideal and need to be higher. The *Energy+FM* system has surpassed the *High-Q* and *Energy only* systems, indicating that the feedback-based adaptive controller has captured additional information using the increased Q-factors, and time-varying Q-factors are more advantageous compared to the fixed high Q-factors. Based on the results in Section 5.1, it can be concluded that the *Energy+FM* system has captured certain artefacts that differentiate replayed signals from genuine ones.

To further analyze the obtained results, trained systems were evaluated on different replay configurations separately, with all 27 acoustic conditions pooled together. Results are shown in Table 4. Since (Alzantot et al., 2019) does not provide results for the evaluation set, only the development set results are presented for the *CQCC-ResNet* model. The first and second letters of the attack type define the attacker-to-speaker distance (D_a) and loudspeaker quality (Q_L), respectively. D_a increases from A to C, whereas Q_L decreases from A to C. Hence, AA is considered the highest-quality attack, and CC is considered the lowest-quality attack. Also, the effect of Q_L is considerably more dominant than that of D_a (Wang et al., 2020). Thus, AA, BA and CA conditions are considered more difficult to detect compared to others. These effects are reflected in the results given in Table 4.

The results indicate that, while the authors' four systems exhibit lower performance on the evaluation set compared to the development set, they display similar trends across systems and attack types. This can be attributed to the fact that both the development and evaluation sets are constructed based on a fixed set of replay categories, with the evaluation set containing replay utterances unseen by the training and development sets. When *Energy+FM* results are compared with the *CQCC-ResNet* baseline, our proposed system outperforms the baseline in most replay configurations. More importantly, when *Energy+FM* is compared with our passive systems and *Energy only* system, *Energy+FM* has outperformed others for the highest quality attacks, which has resulted in overall better performance. Therefore, it is reasonable to presume that the adaptive filter controller has captured artefacts important for detecting higher quality attacks that a non-adaptive model or an energy-based adaptive model was not able to capture.

Fig. 8 illustrates the variation of the Q-factor with time when a genuine signal and two of its spoofed versions are input to the two models: *Energy only* and *Energy+FM*. The two selected spoofing configurations are those of the highest quality in the database. It is seen that for each filter, the *Energy only* model has produced similar Q-factor variations for all three signals. Hence, when the discrimination between two signals is minimal, their subband energy content will be similar, which will result in similar Q-factor variations. In the non-adaptive case as well, all components will be enhanced by a similar amount, and therefore, may not be of much use when the signals are similar. But when the Q-factors produced by the *Energy+FM* model are considered; it is seen that Q-factors of genuine signals are different from the spoofed signals and Q-factors of spoofed signals are different from each other. Therefore, the frequency components highlighted in each signal are different.

The behavior of the three frameworks within the feature space was analyzed by visualizing the neural network features extracted from the second to last fully connected layer (*FC1 back-end*) within a two-dimensional space. Fig. 9 shows the t-SNE plots obtained using all genuine signals and spoofed signals of AA condition of the development set which contains data unseen by the training set. The figure depicts how the proposed framework (Fig. 9(a)) has resulted in relatively better discrimination of genuine and spoofed feature space compared to the

Table 4
Comparison of prediction results (EER(%)) based on replay configurations for the development (Dev) and evaluation (Eval) partitions of ASVspoof 2019 database. Evaluation set results for CQCC-ResNet have not been reported in (Alzantot et al., 2019).

Attack type	Low-Q		High-Q		Energy only		Energy + FM		CQCC – ResNet
	Dev	Eval	Dev	Eval	Dev	Eval	Dev	Eval	Dev (Alzantot et al., 2019)
AA	11.54	12.46	10.45	12.30	12.27	12.52	10.22	12.24	10.59
AB	2.77	6.96	1.46	6.18	1.67	7.31	2.06	6.24	2.57
AC	1.31	2.76	1.26	4.29	0.91	2.97	1.41	3.12	1.75
BA	6.67	8.99	6.72	8.91	7.01	8.66	6.46	8.65	5.46
BB	1.56	4.92	0.94	4.45	1.05	5.00	0.93	4.12	1.18
BC	0.57	1.69	1.04	2.68	0.79	1.95	0.93	2.20	0.84
CA	5.35	7.38	5.93	7.94	6.39	7.34	5.47	7.62	4.55
CB	1.08	3.97	0.82	3.84	0.93	4.58	0.78	3.50	1.00
CC	0.50	1.54	0.78	2.48	0.43	1.78	0.69	1.95	0.82

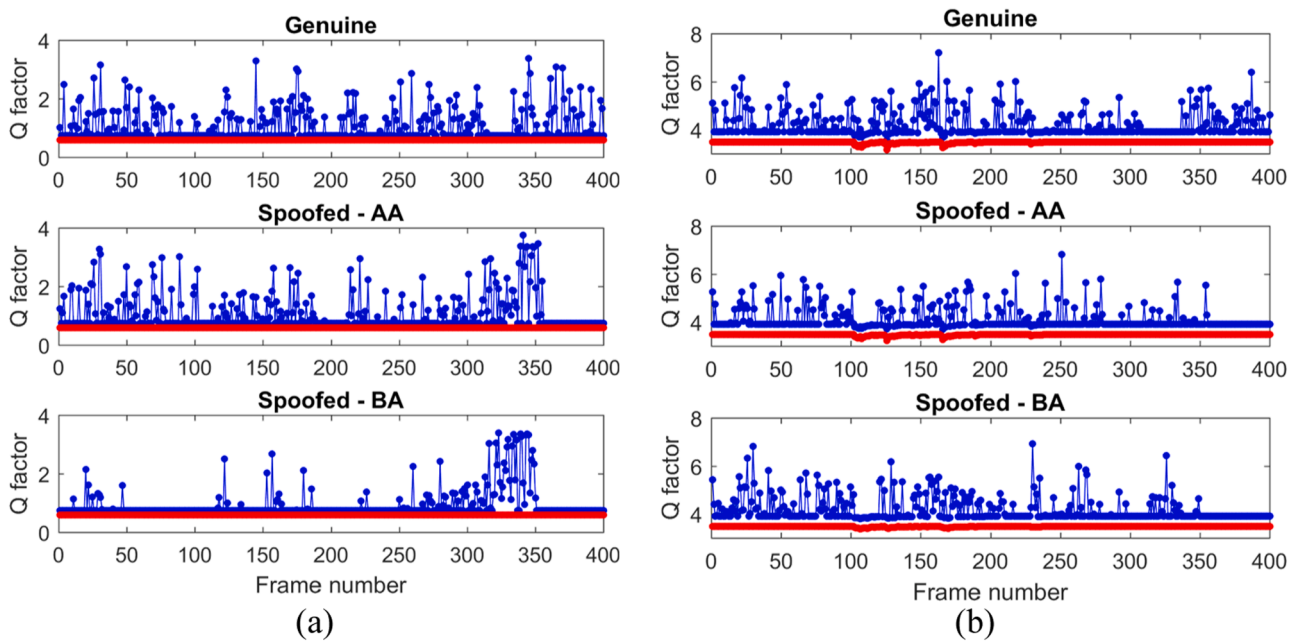

Fig. 8. Variation of Q-factor with time when a genuine signal and two spoofed versions are input to *Energy+FM* (blue) and *Energy only* (red) models for two filters; (a) - Filter 10, and (b) - Filter 36.

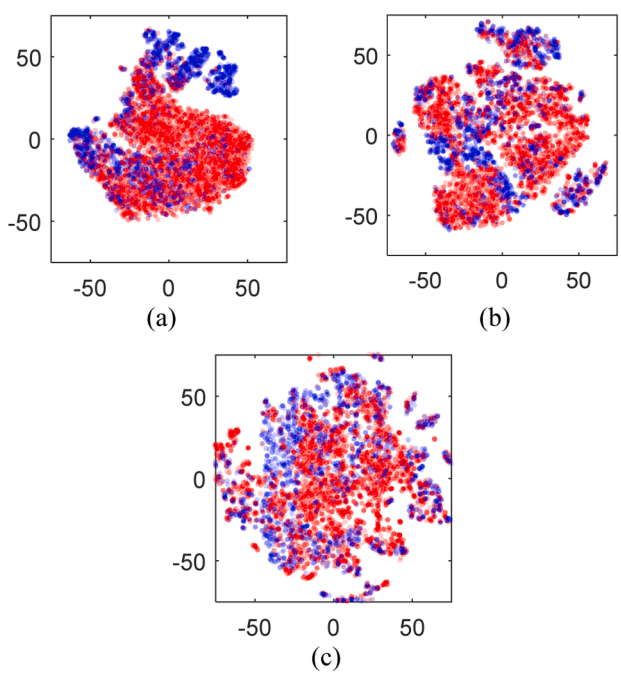

Fig. 9. t-SNE-based two-dimensional space using features extracted from (a) proposed (*Energy+FM*), b) level-dependent framework (*Energy only*) and (c) non-adaptive filter framework (*Low-Q*) for condition AA of the development set of ASVspoof 2019. Genuine features are shown in red and spoofed in blue.

non-adaptive and energy-based frameworks (Fig. 9(b) and (c)). Also, the non-adaptive feature space shows more discrimination compared to the energy-based feature, which is reflected in the results in Table 4 as well. Fig. 9(b) and (c) also indicate more within-class clusters in comparison to the proposed framework. This could mean, replay artefacts are more pronounced in the proposed features while the other two feature sets are affected by variabilities such as speaker characteristics, noise, and speech content.

5.3. Framework analysis on ASVspoof 2021 database

The trained *Energy+FM* system was also tested on the most recently released ASVspoof 2021 database. ASVspoof 2021 serves as an evaluation set, intended to assess the systems trained using the simulated training dataset of ASVspoof 2019. Therefore, the genuine and spoofed data conditions of this database are different from those of ASVspoof 2019 which helps evaluate the generalizability of the system. Table 5 shows the results for the *progress* and *evaluation* phases along with the official baseline systems also trained on ASVspoof 2019 training data. The progress set served as the development set for hyperparameter tuning. Details of the baselines can be found in (Yamagishi et al., 2021). We also evaluated the *CQCC-ResNet* model on ASVspoof 2021 database since other suitable benchmark systems could not be found in the literature. Here, the *CQCC-ResNet* model was trained and evaluated by us using the implementation released by the authors of (Alzantot et al., 2019). As seen on Table 5, the proposed system has outperformed all four baselines and the *CQCC+ResNet* system in both phases. Since all the systems, including our proposed system, were trained on simulated signals and evaluated on real signals, we can conclude that the *Energy+FM* system is able to generalize better when compared to the baseline systems.

6. Conclusion

This paper presents a novel DNN-based adaptive front-end developed based on the human auditory system and applied to replay spoofing

Table 5
Experimental results (EER(%)) as measured on the progress and evaluation phase sets of ASVspoof 2021 database.

System	Progress phase		Evaluation phase	
	EER(%)	t-DCF	EER(%)	t-DCF
CQCC-GMM (Yamagishi et al., 2021)	36.33	0.9033	38.07	0.9444
LFCC-GMM (Yamagishi et al., 2021)	39.79	0.9741	39.54	0.9724
LFCC-LCNN (Yamagishi et al., 2021)	42.16	0.9809	44.77	0.9958
RawNet2 (Yamagishi et al., 2021)	46.03	0.9992	48.60	0.9997
CQCC - ResNet	44.19	0.9798	41.74	0.9587
Energy+FM-ResNet	**33.22**	**0.8394**	**35.26**	**0.8836**

attack detection. Specifically, the proposed model combines input level-dependent filter adaptation, a mechanism known to provide excellent sensitivity and selectivity in the mammalian auditory system, with a DNN-based adaptive filter controller to dynamically generate filters conditioned on the signal energy as well as the spectral characteristics of the outputs of the front-end. The DNN-based adaptive controller was designed to take frame-averaged subband FM as input and jointly optimized with a spoofing detection back-end. Analysis of the features extracted from the framework and replay detection experimental results in terms of EER show that the proposed framework outperforms both conventional (non-adaptive) filtering and input energy-based adaptation when detecting high-quality replay attacks. It is seen that the model can enhance minor differences between genuine and spoofed signals that would otherwise be missed, by appropriately adapting the Q-factors of the front-end filters. The framework's robustness is demonstrated by its ability to generalize to different replay conditions across multiple datasets.

While the proposed framework has shown promising outcomes compared to traditional non-adaptive filter bank-based models and input energy-based adaptive filter models, integration of the proposed framework with more complex back-ends is a promising avenue for future research. Furthermore, the application of the proposed idea to other speech processing tasks also warrants further investigation.

CRediT authorship contribution statement

Buddhi Wickramasinghe: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Visualization. **Eliathamby Ambikairajah:** Conceptualization, Methodology, Resources, Writing – review & editing, Supervision, Funding acquisition. **Vidhyasaharan Sethu:** Conceptualization, Methodology, Resources, Writing – review & editing, Supervision, Funding acquisition. **Julien Epps:** Conceptualization, Resources, Writing – review & editing, Supervision. **Haizhou Li:** Conceptualization, Resources, Writing – review & editing, Supervision. **Ting Dang:** Writing – review & editing, Methodology, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

This work was funded by Australian Research Council (ARC) Discovery grant DP190102479. The authors would also like to thank Data61, CSIRO, Australia for their scholarship support.

Appendix A. Hyperparameters of the framework

The following table summarizes the hyperparameters chosen when training the framework using the proposed front-end features. These hyperparameters were tuned using the development set of the ASVspoof 2019 database.

Table A1
System parameters.

Setting	Hyperparameter value
Feature extraction	
Number of filters in Layer 1	40
Number of filters in Layer 2	39
Number of frames	400
Frame length - Filtering	11 ms
Frame length – Classification	22 ms
DCT coefficients	20
Training	
Optimizer	Adam
Learning rate	0.001
Epochs	35
Loss	Cross-entropy loss

References

Alsteris, L.D., Paliwal, K.K., 2007. Short-time phase spectrum in speech processing: a review and some experimental results. Digit. Signal Process. 17 (3), 578–616.
Alzantot, M., Wang, Z., Srivastava, M.B., 2019. Deep residual neural networks for audio spoofing detection. arXiv preprint arXiv:–1907.00501.
Ambikairajah, E., Kilmartin, L., 1991. An adaptive cochlear model for speech recognition. In: Second European Conference on Speech Communication and Technology.
Australian Taxation Office, 2021. Voice Authentication, Australian Taxation Office. Available at: https://www.ato.gov.au/General/Online-services/Voice-authentication/ (Accessed: 26 August 2023).
Baby, D., Van Den Broucke, A., Verhulst, S., 2021. A convolutional neural-network model of human cochlear mechanics and filter tuning for real-time applications. Nat. Mach. Intell. 3 (2), 134–143.
Belhomme, A., Grenier, Y., Badeau, R., Humbert, E., 2016. Anechoic phase estimation from reverberant signals. In: 2016 IEEE International Workshop on Acoustic Signal Enhancement (IWAENC). IEEE, pp. 1–5.
Darlington, D.J., Campbell, D.R., 1996. Sub-band adaptive filtering applied to speech enhancement. In: Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96, 2. IEEE, pp. 921–924.
De Brabandere, B., Jia, X., Tuytelaars, T., Gool, L.V., 2016. Dynamic filter networks. In: Advances in neural information processing systems, pp. 667–675.
Delcroix, M., Kinoshita, K., Ogawa, A., Huemmer, C., Nakatani, T., 2018. Context adaptive neural network based acoustic models for rapid adaptation. In: IEEE/ACM Transactions on Audio, Speech, and Language Processing, 26, pp. 895–908.
Delgado, H., et al., 2018. ASVspoof 2017 Version 2.0: meta-data analysis and baseline enhancements. In: Odyssey 2018 The Speaker and Language Recognition Workshop, pp. 296–303.
Dinkel, H., Chen, N., Qian, Y., Yu, K., 2017. End-to-end spoofing detection with raw waveform CLDNNS. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*,. IEEE, pp. 4860–4864.
Elliott, S.J., Shera, C.A., 2012. The cochlea as a smart structure. Smart Mater. Struct. 21 (6), 064001.
Evans, N.W., Kinnunen, T., Yamagishi, J., 2013. Spoofing and countermeasures for automatic speaker verification. Interspeech 925–929.

Font, R., Espın, J.M., Cano, M.J., 2017. Experimental analysis of features for replay attack detection–results on the ASVspoof 2017 Challenge. Proc. Interspeech 2017 7–11.
Gao, Y., Lian, J., Raj, B., Singh, R., 2021. Detection and evaluation of human and machine generated speech in spoofing attacks on automatic speaker verification systems. In: IEEE Spoken Language Technology Workshop (SLT), pp. 544–551.
Gomez-Alanis, A., Peinado, A.M., Gonzalez, Gomez, A.M., 2019. A gated recurrent convolutional neural network for robust spoofing detection. IEEE/ACM Trans. Audio Speech Lang. Process. 27 (12), 1985–1999.
Gunendradasan, T., Wickramasinghe, B., Le, N.P., Ambikairajah, E., Epps, J., 2018. Detection of replay-spoofing attacks using frequency modulation features. Proc. Interspeech 2018 636–640.
Gunendradasan, T., Ambikairajah, E., Epps, J., Li, H., 2019. An adaptive-Q cochlear model for replay spoofing detection. INTERSPEECH, pp. 2918–2922.
Hautamäki, R.G., Kinnunen, T., Hautamäki, V., Laukkanen, A.-M., 2015. Automatic versus human speaker verification: the case of voice mimicry. Speech Commun. 72, 13–31.
Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141.
Hussain, A., Campbell, D.R., 1998. Non-linear processing in cochlear spaced sub-bands using artificial neural networks for multi-microphone adaptive speech enhancement. In: 9th European Signal Processing Conference (EUSIPCO 1998). IEEE, pp. 1–4.
Islam, M.A., Jassim, W.A., Cheok, N.S., Zilany, M.S.A., 2016. A robust speaker identification system using the responses from a model of the auditory periphery. PLoS One 11 (7), e0158520.
Kang, D., Dhar, D., Chan, A., 2017. Incorporating side information by adaptive convolution. Adv. Neur. Inform. Process. Syst. 3867–3877.
Kim, S.-H., Park, Y.-H., 2021. Adaptive Convolutional Neural Network for Text-Independent Speaker Recognition. Proc. Interspeech 2021 66–70.
Kinnunen, T., et al., 2017. The ASVspoof 2017 challenge: assessing the limits of replay spoofing attack detection. Interspeech 2–6.
Klein, B., Wolf, L., Afek, Y., 2015. A dynamic convolutional layer for short range weather prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4840–4848.
Kua, J., Epps, J., Ambikairajah, E., Nosratighods, M., 2010. Front-end diversity in fused speaker recognition systems. In: The Proceedings of APSIPA ASC, pp. 14–17.
Lau, Y.W., Wagner, M., Tran, D., 2004. Vulnerability of speaker verification to voice mimicking. In: Intelligent Multimedia, Video and Speech Processing, 2004. Proceedings of 2004 International Symposium on. IEEE, pp. 145–148.
Lavrentyeva, G., Novoselov, S., Malykh, E., Kozlov, A., Kudashev, O., Shchemelinin, V., 2017. Audio replay attack detection with deep learning frameworks. Interspeech 2017 82–86.
Li, X., Wu, X., Lu, H., Liu, X., Meng, H., 2021. Channel-wise gated res2net: towards robust detection of synthetic speech attacks. Proc. Interspeech 4314–4318.
Lindberg, J., Blomberg, M., 1999. Vulnerability in speaker verification-a study of technical impostor techniques. In: Sixth European Conference on Speech Communication and Technology.
Liu, L., He, J., Palm, G., 1997. Effects of phase on the perception of intervocalic stop consonants. Speech Commun. 22 (4), 403–417.
Lyon, R.F., 1990. Automatic gain control in cochlear mechanics. The Mechanics and Biophysics of Hearing. Springer, pp. 395–402.
Maragos, P., Quatieri, T.F., Kaiser, J.F., 1991. Speech nonlinearities, modulations, and energy operators. Proc. ICASSP 91, 421–424.
Nie, K., Stickney, G., Zeng, F.-G., 2004. Encoding frequency modulation to improve cochlear implant performance in noise. IEEE Trans. Biomed. Eng. 52 (1), 64–73.
Biometric Authentication | Strong Customer Authentication | Nuance Australia. Nuance Communications, https://www.nuance.com/en-au/omni-channel-customer-engagement/authentication-and-fraud-prevention/biometric-authentication.html. (Accessed 26 Aug. 2023).
Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., 2019. Pytorch: an imperative style, high-performance deep learning library. Adv. Neural Inf. Process. Syst. 32, 8024–8035.
Patil, H.A., Kamble, M.R., Patel, T.B., Soni, M., 2017. Novel variable length teager energy separation based instantaneous frequency features for replay detection. Proc. Interspeech 2017 12–16.
Ravanelli, M., Bengio, Y., 2018a. Interpretable convolutional filters with sincnet. In: Interpretability and Robustness in Audio, Speech and Language, NIPS 2018 Workshop.
Ravanelli, M., Bengio, Y., 2018b. Speaker recognition from raw waveform with sincnet. In: 2018 IEEE Spoken Language Technology Workshop. SLT, IEEE, pp. 1021–1028.
Riegler, G., Schulter, S., Ruther, M., Bischof, H., 2015. Conditioned regression models for non-blind single image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 522–530.
Sailor, H., Kamble, M., Patil, H., 2018. Auditory filterbank learning for temporal modulation features in replay spoof speech detection. Proc. Interspeech 2018 666–670.
Seki, H., Yamamoto, K., Akiba, T., Nakagawa, S., 2019. Discriminative learning of filterbank layer within deep neural network based speech recognition for speaker adaptation. IEICE Trans. Inf. Syst. 102 (2), 364–374.
Sriskandaraja, K., Sethu, V., Ambikairajah, E., 2018. Deep siamese architecture based replay detection for secure voice biometric. Proc. Interspeech 2018 671–675.
Tak, H., Jung, J.-W., Patino, J., Todisco, M., Evans, N., 2021. Graph Attention Networks for Anti-Spoofing. Proc. Interspeech 2356–2360.
Todisco, M., Delgado, H., Evans, N., 2016. A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients. In: Speaker Odyssey Workshop, Bilbao, Spain, 25, pp. 249–252.
Todisco, M., Delgado, H., Evans, N., 2017. Constant Q cepstral coefficients: a spoofing countermeasure for automatic speaker verification. Comput. Speech Lang. 45, 516–535.
Tom, F., Jain, M., Dey, P., 2018. End-to-end audio replay attack detection using deep convolutional networks with attention. Proc. Interspeech 2018 681–685.
Van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn Res. 9 (11).
Vincent, E., 2016. Roomsimove. Available at: http://homepages.loria.fr/evincent/software/Roomsimove_1.4.zip (Accessed 05/08/2020).
Wan, G., Pan, J., Wang, Q., Gao, J., Ye, Z., 2020. Speaker adaptive training for speech recognition based on attention-over-attention mechanism. INTERSPEECH 1251–1255.
Wang, X., et al., 2020. ASVspoof 2019: a large-scale public database of synthesized, converted and replayed speech. Comput. Speech Lang., 101114
Wickramasinghe, B., Ambikairajah, E., Epps, J., 2019a. Biologically inspired adaptive-Q filterbanks for replay spoofing attack detection. In: Proc. Interspeech 2019, pp. 2953–2957.
Wickramasinghe, B., Ambikairajah, E., Epps, J., Sethu, V., Li, H., 2019b. Auditory inspired spatial differentiation for replay spoofing attack detection. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 6011–6015.
Witkowski, M., Kacprzak, S., Zelasko, P., Kowalczyk, K., Gałka, J., 2017. Audio replay attack detection using high-frequency features. Proc. Interspeech 27–31, 2017.
Wu, Z., Li, H., 2013. Voice conversion and spoofing attack on speaker verification systems. In: Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific. IEEE, pp. 1–9.
Wu, Z., Gao, S., Cling, E.S., Li, H., 2014. A study on replay attack and anti-spoofing for text-dependent speaker verification. In: Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA). IEEE, pp. 1–5.
Wu, Z., Evans, N., Kinnunen, T., Yamagishi, J., Alegre, F., Li, H., 2015. Spoofing and countermeasures for speaker verification: a survey. Speech Commun. 66, 130–153.
Yamagishi, J., et al., 2021. ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection. Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, pp. 47–54.
Zeinali, H., et al., 2019. Detecting spoofing attacks using vgg and sincnet: but-omilia submission to asvspoof 2019 challenge. INTERSPEECH.
Zhang, Q., Song, Q., Nicolson, A., Lan, T., Li, H., 2021. Temporal convolutional network with frequency dimension adaptive attention for speech enhancement. Proc. Interspeech 2021 166–170.