ARawNet: A Lightweight Solution for Leveraging Raw Waveforms in Spoof Speech Detection

Zhongwei Teng*, Quchen Fu*, Jules White*, Maria E. Powell† and Douglas C. Schmidt*
*Dept. of Computer Science, Vanderbilt University
†Dept. of Otolaryngology–Head and Neck Surgery, Vanderbilt University Medical Center

***Abstract*—An emerging trend in audio processing is capturing low-level speech representations from raw waveforms. These representations have shown promising results on a variety of tasks, such as speech recognition and speech separation. Compared to handcrafted features, learning speech features via backpropagation can potentially provide the model greater flexibility in how it represents data for different tasks. However, results from empirical studies show that, in some tasks, such as spoof speech detection, handcrafted features still currently outperform learned features. Instead of evaluating handcrafted features and raw waveforms independently, this paper proposes an Auxiliary Rawnet model to complement handcrafted features with features learned from raw waveforms for spoof speech detection. A key benefit of the approach is that it can improve accuracy at a relatively low computational cost. The proposed Auxiliary Rawnet model is tested using the ASVspoof 2019 dataset and pooled EER and min-tDCF are 1.11% and 0.03645 respectively. Results from this dataset indicate that a lightweight waveform encoder can boost the performance of handcrafted-features-based encoders for 10 types of spoof attacks, including 3 challenging attacks, in exchange for a small amount of additional computational work.**

***Index Terms*—Raw waveform, handcrafted features, spoof speech detection**

I. Introduction

Fixed, handcrafted audio features, such as Mel-filter banks [1], have shown great performance in capturing strong audio features in aspects of both auditory and machine learning [2], [3]. However, since handcrafted features are often designed based on specific tasks, such as speech recognition, using these features to solve problems that they were not designed for may not be optimal. For example, Mel-filter banks [1] apply triangular filter banks on a Mel-scale to spectrograms calculated using short-term Fourier transform (STFT) to represent the non-linear perception of the human hearing. The Mel-scale is derived from a set of perception experiments on humans. As a result, Mel-filter banks are coarse-grained at high-frequencies since humans are less sensitive to high frequency sound. This loss of signal energy (information) in high frequencies may lead to poor performance on tasks that rely on information in these higher frequencies [3].

Extracting audio features with backpropagation provides an alternative way to represent raw waveforms by using deep neural networks to learn task-specific features. Task-specific features can be learned for many problems, such as voice recognition [4], [5] or automatic speaker verification (ASV) [6]. Directly learning features from raw waveforms grants greater flexibility in handling unknown tasks and, thus, overcomes some of the challenges of handcrafted features, which may lose signal energy needed by a specific task. Previous research indicates that representations learned from waveforms still have limitations on signal energy loss compared to the original raw signals they were learned from [2]. In the spoof speech detection task, models based on only raw waveforms perform better in specific spoof attacks, while shows weaker performance on other attacks compared to model based on handcrafted data [7], [8].

Instead of relying on raw waveforms independently, a potential option is to design a solution that can take advantage of both handcrafted and learned features. For example, lost phase information in handcrafted features can be complemented by features learned from raw waveforms. However, building an end-to-end CounterMeasure (CM) systems, containing multiple encoders to process raw waveforms and hand-crafted features independently, can result in big challenges in model size and complexity.

In this paper, we propose the Auxiliary Rawnet (ARNet) architecture to combine learned features from raw waveforms with existing handcrafted features, by designing a lightweight auxiliary encoder. The proposed model was tested on the ASV Spoof 2019 dataset [9], where the model needs to defend against speech spoofing attacks from a variety of sources. The model shows promise in boosting the performance of handcrafted feature-based networks that warrants further investigation on additional data sets and tasks.

The key contributions of this paper are as follows:

- We elaborate on the problem of concatenating raw waveforms and handcrafted features in the speech field and propose assumptions to solve this problem efficiently.
- Based on our assumption, we introduce the Auxiliary Rawnet architecture that can be used to attach a lightweight auxiliary encoder to a model that relies on handcrafted features, so that raw waveform data can supplement the information in handcrafted features.
- We show results that indicate that, by introducing the auxiliary raw encoder, model performance is boosted on the ASV spoof 2019 dataset. A light-weight auxiliary encoder boosts model performance over 10 of 13 spoof attacks, including 3 challenging attacks.
- We describe how our results show the potential of combining a light-weight waveform encoder with other encoders, providing an approach to balance the trade-off

between performance and model complexity for models containing multiple encoders.

The remainder of this paper is organized as follows: SectionII discusses prior work in audio signal feature representation. SectionIII explains the problem analyzed in this paper and describes the Auxiliary Rawnet structure. SectionIV introduces the experimental dataset and tasks used in this paper. Experimental results are analyzed in SectionV. SectionVI presents concluding remarks and lessons learned.

II. Related Work

Prior work has shown how the ”front-end” of models, which extract features from raw data, can be improved by using deep neural networks [2], [3], [5], [10], [11] to directly learn features from raw signal data. Directly applying standard CNNs to process raw waveforms [12] has shown promising results in speech recognition, spoofing detection, and speech separation. Convolutions on time-domain raw waveforms can be explained as finite impulse response filter banks [2]. Structured filters are applied to optimize standard CNNs based on digital signal processing theory, by initializing the first convolutional layer, which is believed to be the most important part, with known filter families [11], [13], [14], so that a custom filter bank can be designed for a specific task. Filter-based waveforms networks are emerging as excellent front-ends for many tasks [3], [7]. However, a theoretical analysis from Joakim et al. [2] has shown that signal energy loss is still inevitable for features extracted from raw waveforms by a CNN. Their results show extracted features can carry up to 94.5% signal energy compared to the original waveforms. On the other hand, empirical research also indicates that handcrafted features are still competitive in specific questions, such as speech commands [3], spoof speech detection [9], [15], and instrument classification [3].

In prior work on spoof detection, a deep neural network, called RawNet2 [7], used raw waveforms to enhance the performance of CM systems against certain types of spoof attacks, but at the expense of increased model size and computational complexity. The prior work showed that models relying solely on raw waveforms showed weaker performance on many types of spoof attacks, resulting in worse overall performance on spoof detection according to the ASV spoof dataset [7]. It is challenging to create an end-to-end network, which can take both raw waveforms and hand-crafted features as input due to the increase in model size that accompanies raw waveform use. This paper considers the use of raw waveforms as a supplement to handcrafted features, rather than the main input, for spoof detection and investigates their potential to boost performance with little additional computational cost. To the best of our knowledge, our work is the first study on the bottleneck structure of hand-crafted features and raw waveforms in deep learning models for spoof detection. The paper also presents the first architecture to apply both hand-crafted features and raw waveforms in an end-to-end model for spoof detection tasks.

III. Auxiliary RawNet

This section elaborates the research problem on combining raw waveforms and handcrafted features and explains the structure of the ARNet architecture.

The proposed network architecture applies a light-weight encoder to process raw waveforms with low computational cost as learned features, which are combined with existing speech classification models (Figure 1). To produce disentangled representations from different encoders, a narrow bottleneck is leveraged in the raw waveforms encoder without damaging the performance of the handcrafted encoder, as shown in Figure 2.

A. Problem Formulation

Before introducing the ARNet architecture, we first formalize the problem that it is intended to solve. Denote F_w as features of a raw waveform, and p as a problem to solve. We assume there is a constructive function f, which can map $F_{p_{mag}}$, $F_{p_{phase}}$ and $S_{p_{noise}}$ into F_w, as described in Equation 1, where $F_{p_{mag}}$ is the ideal magnitude information needed to solve p, $F_{p_{phase}}$ is the ideal phase information needed to solve p, and $S_{p_{noise}}$ are signals with limited contribution to solving p (e.g., background noise).

$$F_w = f(F_{p_{mag}}, F_{p_{phase}}, S_{p_{noise}}) \tag{1}$$

Empirical studies [3] have shown the ability of handcrafted features to represent the strongest audio features for a variety of problems. Based on our assumption, the calculation of handcrafted features can be denoted as a mapping function g, which can retrieve approximations of $F_{p_{mag}}$ or $F_{p_{phase}}$. For example, mel-spectragrams can be described by the following equation:

$$F_{p_{mag}} \approx F_{mel} = g_{mel}(|STFT(F_w)|^2)) \tag{2}$$

When concatenating raw waveform data and handcrafted features to enhance model performance, our work is essentially to find a function, h, so that the total loss of $g(F_w)$ and $h(F_w)$ is smaller than a single $g(F_w)$. In other words, we want to find representations closer to the ideal solution $F_{p_{mag}} + F_{p_{phase}}$, as describe in Equation 3.

$$concat(g(F_w), h(F_w)) \approx F_{p_{mag}} + F_{p_{phase}} > g(F_w) \tag{3}$$

However, it is not clear how $g(F_w)$ interacts with $h(F_w)$. Inspired by observations from results regarding $g(F_w)$ and $h(F_w)$ on various tasks [3], [7], we make the following assumption about combining learned features and handcrafted features:

Assumption 1 (A1): *If a handcrafted feature, $g(F_w)$ shows strong results solving problem p, then there exists a $h(F_w)$ with size less than N in $concat(g(F_w), h(F_w))$ that will enhance overall performance. In other words, $h(F_w)$ can be an auxiliary component of $g(F_w)$ to improve performance with a bounded cost.*

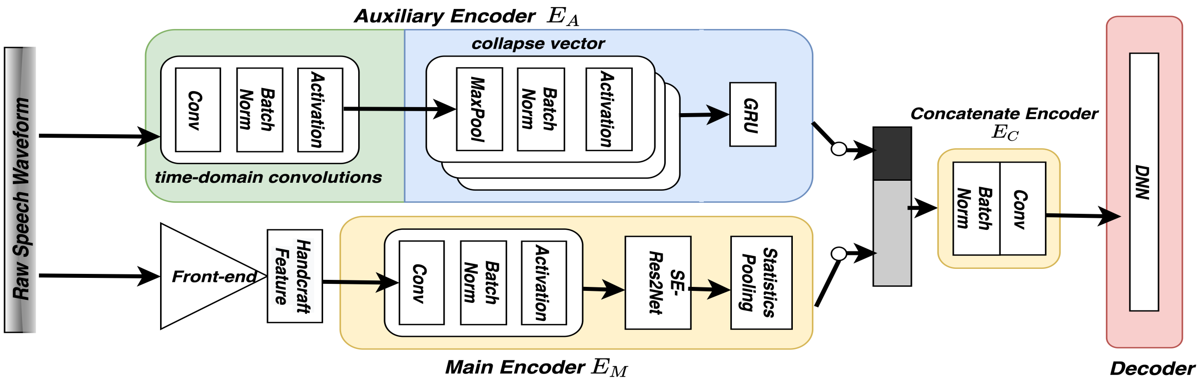

Fig. 1. The ARNet Architecture. E_A contains one strided CNN, 3 continuous max-pooling layers and a GRU. A TDNN-based model is illustrated here as an example of the E_M.

B. The Auxiliary RawNet Structure

Based on the assumptions presented in section III-A, we propose the ARNet architecture. An overview of the ARNet architecture is shown in Figure 1. E_A, which processes the raw waveform, has a smaller bottleneck than E_M which processes handcrafted audio features, to make the raw waveforms play a supplementary role and bound the computational cost (e.g., bound N).

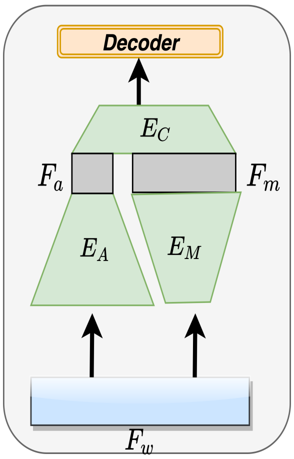

Fig. 2. Overview of the ARawNet Architecture. The model consists of a Main Encoder(E_M), Auxiliary Encoder(E_A), and Concatenate Encoder(E_C). E_A has a smaller bottleneck than E_M.

The Encoders. There are 3 encoders in the ARNet: the Main Encoder(E_M), Auxiliary Encoder(E_A), and Concatenate Encoder(E_C). E_M denotes the main encoder, whose inputs are the original handcrafted features that have shown good performance in solving the target problem. E_A is the encoder used to encode the raw waveforms in a light-weight way to compress F_w into F_a, where F_a are the features extracted by the auxiliary encoder. F_a and F_m (hand crafted features from the main encoder) are then concatenated in channels and further encoded by E_C.

Figure 1 shows details of the encoders used in our experiments on the ASVspoof 2019 dataset. We select the strided convolutional layer [5] as the first layer to directly process the raw waveforms. However, unlike previous raw waveforms networks, which include multiple CNN blocks with large kernels, the strided convolutional layer is only followed by 3 continuous pooling blocks to collapse vectors and remove any frame variance without further convolution. A GRU is used to encode frame-level features into utterance-level embeddings by keeping output vectors from the last time step.

The main encoder keeps layers before the statistical pooling layer, which will output utterance-level embeddings. Based on our assumption 1, we chose a narrow bottleneck for E_A. The dimension of the utterance-level embedding from E_A is designed to be smaller than the output dimension from E_M. In the end, E_C only contains a single Conv1d to encode concatenated results from E_A and E_M.

The full architecture and model hyper-parameters are explained in Table I.

Encoders	Blocks
Auxiliary Encoder	Conv(3,3,128) BN&LeakyReLu MaxPooling BN&LeakyReLu GRU(512)
Concatenate Encoder	BN Conv1D(1,1,256)

TABLE I
THE ARCHITECTURE OF AUXILIARY ENCODER AND CONCATENATE ENCODER.

The Decoder. In our problem, the decoder is a linear classifier layer that decodes embeddings from E_C to target classification.

C. Why does a light-weight encoded raw waveforms augment handcrafted features?

(1) Compared to the current filter-based architectures as discussed in Section II, we chose the strided convolutional receptive field, which is a standard CNN, as the first layer to process the raw waveforms. The strided convolutional layer consists of a set of time-domain convolutions, where all parameters(CNN kernel), are learned from the data. Calculation of the first CNN layer can be described as the following Equation [11], where x[n] is raw waveforms, h[n] is the filter and y[n] is filtered output:

$$y[n] = x[n] * h[n] = \sum_{0}^{L-1} x[l] \cdot h[n-l] \tag{4}$$

As discussed in Section III-A, concatenating $g(F_w)$ and $h(F_w)$ requires each encoder to have different attention to features in the raw waveforms so that they can complement each other. The standard convolutional layer with small kernels gives the E_A the least information about the signal processing mechanisms in $g(F_w)$, and thus potentially grants it the most flexibility to extract features, which do not overlap with $g(F_w)$.

(2) In contrast to previous waveform-based networks [5], [7], the CNN blocks used in between the strided convolution layer and the GRU are completely removed, and only 3 continuous max-pooling layers with batch normalization are kept to collapse frame-level features step-by-step.

The first convolutional layer is considered the most critical part in processing raw waveforms. In deep networks it is also the most vulnerable to problems, such as vanishing gradients, without initializing filters [11]. However, based on our assumption 1, only significant frame-level features need to be kept, indicating networks without deep CNN blocks can be used for E_A. Max pooling layers are used to collapse vectors and find significant pattern information that can be visualized after 3 pooling layers, as shown in Figure 3.

(3) We test our assumption 1 based on the Theorem [16] from speech conversion problems, that if information bottlenecks between different encoders are precisely set, the model will decompose and produce disentangled representations of input speech signals. In our model, this Theorem can be described by the following equation:

$$E_M(F_w) = g(F_w), E_A(F_w) = h(F_w) \tag{5}$$

Thus, a narrow bottleneck is designed for E_A, which means the dimension of utterance-level embeddings dim_{E_A} is much smaller than dim_{E_M}.

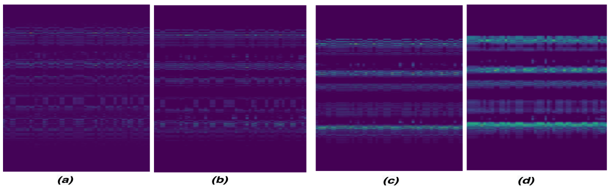

Fig. 3. Outputs visualization of the strided convolution layer and pooling layers. Outputs after 3 pooling layers(d) shows signification pattern information.

(4) Output embeddings from E_M and E_A are concatenated in the utterance level, where segment-level layers in single encoder are removed and are replaced with concatenate encoder E_C. This is a critical features for the ARawNet, as well as for our assumption 1, that original bottleneck layers from E_M and E_A should be replaced with a E_C. We concluded the following hypothesis as we designed the network:

Assumption 2 (A2): *Given raw waveform F_w and handcrafted features $h(F(w))$, segment-level layers need to be designed after the concatenating layer to represent disentangled representations.*

IV. Experimental Setup

A. Experimental Dataset

The ASVspoof 2019 logical access (LA) dataset was developed to improve research on the growing threat of voice spoofing attacks on automated speech verification systems [9]. This dataset contains human-recorded audios and spoof audios generated from 19 sources (A01 - A19), including speech synthesis, voice conversion, and hybrid algorithms.

50,224 records in the training and development data consist of spoof attacks generated by A01-A06. Another 71,237 spoof audio files in the evaluation data are generated by A07-A19, which are unpredictable spoofing attacks for CounterMeasure (CM) systems. Detailed statistics of the ASVSpoof 2019 is shown in Table II.

Subsets	#Bonafide	#Spoofed	Spoof Source
Training	2580	22800	A01-A06
Development	2548	22296	A01-A06
Evaluation	7355	63882	A07-A19

TABLE II
STATISTICS OF THE ASV2019

We chose the ASVspoof 2019 LA dataset to validate the performance of our proposed model since:

- The performance of handcrafted features is limited by the difference in spoofing sources between the training and evaluation data. Spoofing types are highly unpredictable while the performance of CM systems relies on known spoofing attacks in training data and shows worse performance on unknown spoof attacks.
- Current results on the ASVspoof 2019 challenge [7], [9] indicate that correct handcrafted features still provide the most competitive results from a single model compared raw waveforms approaches.
- Waveforms-based network outperforms on specific types of spoof attacks with worse pooling results compared to handcrafted feature based networks [7].

B. Evaluation Metrics

Two metrics are used to evaluate the ASVspoof 2019 LA dataset including *min t-DCF* as the primary metric and $EqualErrorRate(EER)$ as a secondary metric, as described in [9]:

1) min t-DCF: The Tandem Detection Cost Function (t-DCF) [17] extends the conventional Detection Cost Function (DCF) in voice verification systems for spoofing attacks. The t-DCF measures the overall effect of CM systems combined with existing ASV systems. The CM system acts as a gateway for the ASV system and this metric measures the overlapping of the two, a smaller value indicates better protection against spoofing. min t-DCF can be calculated in Equation 6 [17], where $P^{\mathrm{cm}}_{\mathrm{miss}}(s)$ is the CM miss and $P^{\mathrm{cm}}_{\mathrm{fa}}(s)$ is the false alarm rates.

$$\mathrm{t-DCF}^{\mathrm{min}}_{\mathrm{norm}} = \min_{s} \{\beta P^{\mathrm{cm}}_{\mathrm{miss}}(s) + P^{\mathrm{cm}}_{\mathrm{fa}}(s)\} \tag{6}$$

Spoof Attack [9]			EER			min-tDCF		
	Category	Acoustic mode	CQT+Aux	CQT	Effect(%)	CQT+Aux	CQT	Effect(%)
A7	TTS	LSTM-RNN	0.28521	0.3667	22.2	0.00895	0.01168	23
A8	TTS	AR LSTM-RNN	2.4786	1.72823	-43.4	0.07071	0.04895	-44
A9	TTS	LSTM-RNN	0.08149	0.146	44.1	0.0022	0.00424	48
A10	TTS	Attention seq2seq model	**0.46516**	**0.79111**	**41.2**	**0.01351**	**0.02326**	**41**
A11	TTS	Attention seq2seq model	0.30218	0.77414	60.9	0.00951	0.02527	62
A12	TTS	-	0.24447	0.3667	33.3	0.00703	0.01226	42
A13	TTS-VC	Moment matching NN	**0.12223**	**0.22069**	**44.6**	**0.00376**	**0.00702**	**46**
A14	TTS-VC	LSTM-RNN	0.12223	0.28521	57.1	0.00376	0.00874	56
A15	TTS-VC	LSTM-RNN	0.30218	0.5127	41	0.00907	0.01654	45
A16	TTS	-	0.24447	0.22069	-10.7	0.00768	0.00649	-18
A17	VC	VAE	3.11354	2.40391	-29.5	0.09041	0.07587	-19
A18	VC	i-vector/PLDA	**0.8726**	**5.28996**	**83.5**	**0.02837**	**0.16156**	**82**
A19	VC	GMM-UBM	1.05935	2.09493	49.4	0.0361	0.06432	43

TABLE III

MODEL PERFORMANCE BY SPOOF CATEGORY

2) EER: EER indicates the threshold of a CM system where the false positive and false negative rates are equal each to other.

C. Hand-crafted feature selection

To validate assumption 1, as well as the performance of our ARawNet implementation, hand-crafted features need to be selected carefully in our experiments. According to our assumption, it is important to ensure that the selected hand-crafted features show strong results on the target dataset, so that we can use a lightweight network to enhance the overall performance without a large increase in network complexity.

To reduce the performance bias of a single hand-crafted feature in our experiment and choose appropriate features, we carefully reviewed prior work on the ASVspoof 2019 challenge and chose the state-of-the-art model as our benchmark, so that the features we selected have validated performance on the ASVspoof 2019 dataset. By choosing validated hand-crafted features with state-of-the-art performance, we can be more confident in the analysis and exploration of our assumption on augmentation with raw waveforms.

Specifically, 3 hand-crafted features were selected in our experiments as described below:

- **Mel-Spectrogram**. Mel-Spectrogram is a Short-Time Fourier Transform (STFT) based frontend, which is used to represent the non-linearity of the human ear's sensitivity to different frequencies by applying filters in a mel-scale, as shown in Equation 7.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{7}$$

- **Mel-frequency cepstral coefficients(MFCC)**. MFCC is a popular feature for speech recognition, which is a set of coefficients of the mel-frequency cepstrum. MFCC is calculated by applying Discrete Cosine Transform (DCT) to Mel-Spectrograms to decorrelate them.
- **Constant Q Transform (CQT)**. CQT is a feature to improve downstream tasks, where mel-scale does not perform well, such as music recognition. The CQT center frequency calculation is shown in Equation 8 [18].

$$f_k = f_0 2^{\frac{k}{B}} \tag{8}$$

D. Baseline Setup

Our experiments include one handcrafted feature-based system and one raw waveform-based system respectively:

- **Res2net Architecture**. The Res2net architecture [8] is the state-of-the-art single system in the ASVspoof 2019 challenge, which tested the performance of 3 handcrafted features: log power magnitude spectrogram (Spec), linear frequency cepstral coefficients (LFCC), and constant-Q transform (CQT).
- **RawNet2**. The RawNet2 [7] is the first anti-spoofing model, which only relies on the raw waveforms as input.

V. RESULTS AND ANALYSIS

Table IV shows the experimental results of the ARawNet on the ASVSpoof 2019 dataset.

Front-end	Main Encoder	E_A	EER	min-tDCF	Improve
Spec [8]	Res2Net [8]	-	8.783	0.2237	
LFCC [8]		-	2.869	0.0786	
CQT [8]		-	2.502	0.0743	
Raw waveforms [7]	Rawnet2 [7]	-	5.13	0.1175	
Mel-Spectrogram	XVector	✓	**1.32**	0.03894	43%
		-	2.39320	0.06875	
Mel-Spectrogram	ECAPA-TDNN	✓	**1.39**	0.04316	32.8%
		-	2.11	0.06425	
CQT	XVector	✓	**1.74**	0.05194	45.3%
		-	3.39875	0.09510	
CQT	ECAPA-TDNN	✓	**1.11**	0.03645	28.2%
	.	-	1.72667	0.05077	
MFCC	XVector	✓	**1.39**	0.03830	45.1%
	.	-	2.45	0.06981	
MFCC	ECAPA-TDNN	✓	**1.33**	0.04260	37.7%
	.	-	2.41	0.06838	

TABLE IV

RESULTS ON THE ASVSPOOF 2019 DATASET

- **Results demonstrate the effectiveness of adding a light-weight auxiliary encoder to the main encoder.** Three handcrafted features, Mel-spectrogram, CQT [18] and MFCC, as well as two state-of-the-art models in the speaker verification problem (XVector [19], [20] and ECAPA-TDNN [20], [21]) are selected as main encoders in the ARNet architecture. Without modifying the hyper-parameters in the main encoder, we add the auxiliary encoder, as described in Table I, in the network to evaluate our assumption. Overall, by introducing the auxiliary encoder, both pooled EER and $min - tDCF$ are reduced by $\tilde{5}0\%$ in all combinations of front-end and main encoders. Specifically, CQT/ECAPA-TDNN with auxiliar encoder reaches the best performance on EER of 1.11% and $min - tDCF$ of 0.0364, which is reasonable since single CQT perform best in given benchmark [8].

- **Performance of the CM system on challenging attacks (A10, A13, and A18) is boosted when using CQT as an input feature.** Table III explains the category of different attacks in the evaluation set and breaks down the performance of our model in a different subset of spoof attacks. Overall, detection of 10 of 13 spoof attacks was improved with the lightweight auxiliary model, and the auxiliary model boosts detection of most attacks generated from LSTM-based models. Among those attacks, A10 [22], A13 [23], and A18 [24] are considered as high risks to ASV systems as well as challenging attacks for CM systems to detect [9]. As shown in Table III, the ARawNet largely decreased the EER and the min-tDCF for A10, A13, and A18. Especially for attack A18, model performance improves by over 80%, the EER reduced from 5.29% to 0.87% and min-tDCF reduced from 0.162 to 0.028. However, we also noticed that there are negative effects of including a light-weight raw waveform encoder on certain spoof attacks. For example, the A17 attack, which is generated by variational autoencoder(VAE) [25], is hard for CM systems to detect even though it is a minor threat to the ASV system [9]. Prior works show that a raw waveform-based network with deep layers shows a better performance on this type of attack [7]. This result indicates that we may need to enlarge the network to enhance model performance on specific types of spoof attacks.
- **By introducing the lightweight raw-waveform encoder, the model is less sensitive to highly non-linear information from hand-crafted features.** Mel-spectrograms have become more popular than MFCCs in recent research, since deep neural networks are less likely to be weakened by highly correlated input. After whitening the mel-spectrogram, highly non-linear information is removed in MFCCs, which may be useful for networks. Results in our experiments show that, without raw waveforms, mel-spectrogram shows better performance than the MFCC, indicating that highly non-linear information plays an important role in recognizing spoof attacks. While results of both features tend to be similar after introducing the lightweight raw-waveform encoder. It implies that the auxiliary encoder disentangles raw waveforms and helps the model complement lost highly correlated information in MFCCs.
- **Improper concatenation of the raw waveform encoder output and hand-crafted encoder output can lead to worse results.** Table V shows the performance comparison of models with different concatenating strategies. As we discussed in hypothesis 2, in spoof attack tasks, segment-level layers need to be placed after concatenating utterance-level features from different encoders. Using CQT as features and ECAPA-TDNN as the main encoder, we adjust the model by concatenating segment-level features rather than utterance-level features. Results shows EER increased by 36% with 45% increment on the trainable parameters.

Features	Concatenate	EER	min-tDCF	Parameters
CQT	Before segment layer	1.11	0.03645	10.4 M
	After segment layer	1.36	0.04052	7.18 M

TABLE V
MODEL PERFORMANCE WITH DIFFERENT CONCATENATING LAYERS

- **ARawNet adds raw waveform information to the spoof detection tasks with a smaller network size than prior approaches.** Table VI compares the number of trainable parameters, model complexity, and multiply-and-accumulates (MACs) in our experiments. Compared to encoding handcrafted features (Res2Net), directly encoding raw waveforms (Rawnet2) increases model size and complexity by 2400% and 600%. On the other hand, our auxiliary waveforms encoder only takes up 1.15M trainable parameters, which is a 19% increase in ECAPA-TDNN and the model complexity increases from 2.36 GMac to 3.19 GMac. In other words, the performance of our model increases by 28.2% with increments of 35.1% MACs.
  The smaller network size allows us to train the model on 2 Nvidia 2080 Ti GPUs with the batch size set to 24.

Main Encoder	Auxiliary Encoder	Parameters	MACs
Rawnet2	-	25.43 M	7.61 GMac
Res2Net	-	0.92 M	1.11 GMac
XVector	✓	5.81 M	2.71 GMac
XVector	-	4.66M	1.88 GMac
ECAPA-TDNN	✓	7.18 M	3.19 GMac
ECAPA-TDNN	-	6.03M	2.36 GMac

TABLE VI
COMPARISON OF MODEL COMPLEXITY (MACS) OF VARIOUS SPOOF DETECTION SYSTEMS

VI. CONCLUSION AND FUTURE WORK

This paper discussed the problem of combining learned and handcrafted features to build deep neural networks for the spoof voice detection task. Based on our assumption that hand-crafted features and raw waveforms may complement each other without sacrificing model complexity, we investigated the concatenation of multiple encoders and proposed ARawNet, which includes both hand-crafted features and raw waveforms as inputs, while maintaining a relatively small network size. We tested 3 hand-crafted features (Mel-spectrogram, MFCC, an CQT) and 2 state-of-the-art models (XVector and ECAPA-TDNN) as the main encoder with our Auxiliary Encoder. Experiment results show raw waveforms have the ability to complement CQT for detection of most spoof attacks in the ASVspoof 2019 dataset, as well as its ability to complement the highly non-linear information for MFCC features. The code described here is available in open-source form from: `github.com/magnumresearchgroup/AuxiliaryRawNet`.

References

[1] G. Fechner, "Elements of psychophysics. vol. i." 1966.

[2] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4114–4128, 2014.

[3] N. Zeghidour, O. Teboul, F. d. C. Quitry, and M. Tagliasacchi, "Leaf: A learnable frontend for audio classification," *arXiv preprint arXiv:2101.08596*, 2021.

[4] J.-W. Jung, H.-S. Heo, I.-H. Yang, H.-J. Shim, and H.-J. Yu, "Avoiding speaker overfitting in end-to-end dnns using raw waveform for text-independent speaker verification," *extraction*, vol. 8, no. 12, pp. 23–24, 2018.

[5] J.-w. Jung, H.-S. Heo, J.-h. Kim, H.-j. Shim, and H.-J. Yu, "Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification," *arXiv preprint arXiv:1904.08104*, 2019.

[6] A. T. Liu, S.-W. Li, and H.-y. Lee, "Tera: Self-supervised learning of transformer encoder representation for speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[7] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6369–6373.

[8] X. Li, N. Li, C. Weng, X. Liu, D. Su, D. Yu, and H. Meng, "Replay and synthetic speech detection with res2net architecture," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6354–6358.

[9] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. Kinnunen, and K. A. Lee, "Asvspoof 2019: Future horizons in spoofed and fake audio detection," *arXiv preprint arXiv:1904.05441*, 2019.

[10] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[11] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 1021–1028.

[12] D. Palaz, M. M. Doss, and R. Collobert, "Convolutional neural networks-based continuous speech recognition using raw speech signal," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4295–4299.

[13] P.-G. Noé, T. Parcollet, and M. Morchid, "Cgcnn: Complex gabor convolutional neural network on raw speech," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7724–7728.

[14] R. Balestriero, R. Cosentino, H. Glotin, and R. Baraniuk, "Spline filters for end-to-end deep learning," in *International conference on machine learning*. PMLR, 2018, pp. 364–373.

[15] Q. Fu, Z. Teng, J. White, M. E. Powell, and D. C. Schmidt, "Fastaudio: A learnable audio front-end for spoof speech detection," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 3693–3697.

[16] K. Qian, Y. Zhang, S. Chang, M. Hasegawa-Johnson, and D. Cox, "Unsupervised speech decomposition via triple information bottleneck," in *International Conference on Machine Learning*. PMLR, 2020, pp. 7836–7846.

[17] T. Kinnunen, K. A. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "t-dcf: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," *arXiv preprint arXiv:1804.09618*, 2018.

[18] C. Schörkhuber and A. Klapuri, "Constant-q transform toolbox for music processing," in *7th sound and music computing conference, Barcelona, Spain*, 2010, pp. 3–64.

[19] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[20] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "SpeechBrain: A general-purpose speech toolkit," 2021, arXiv:2106.04624.

[21] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[22] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. L. Moreno *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *arXiv preprint arXiv:1806.04558*, 2018.

[23] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *International Conference on Machine Learning*. PMLR, 2015, pp. 1718–1727.

[24] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector plda: Towards unifying speaker verification and transformation," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 5535–5539.

[25] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," *arXiv preprint arXiv:1704.00849*, 2017.