



# UR Channel-Robust Synthetic Speech Detection System for ASVspoof 2021

Xinhui Chen\*, You Zhang\*, Ge Zhu\*, Zhiyao Duan

Audio Information Research Lab, University of Rochester, Rochester, NY, USA

you.zhang@rochester.edu

## Abstract

In this paper, we present UR-AIR system submission to the logical access (LA) and the speech deepfake (DF) tracks of the ASVspoof 2021 Challenge. The LA and DF tasks focus on synthetic speech detection (SSD), i.e. detecting text-to-speech and voice conversion as spoofing attacks. Different from previous ASVspoof challenges, the LA task this year presents codec and transmission channel variability, while the new task DF presents general audio compression. Built upon our previous research work on improving the robustness of the SSD systems to channel effects, we propose a channel-robust synthetic speech detection system for the challenge. To mitigate the channel variability issue, we use an acoustic simulator to apply transmission codec, compression codec, and convolutional impulse responses to augmenting the original datasets. For the neural network backbone, we adopt Emphasized Channel Attention, Propagation and Aggregation Time Delay Neural Networks (ECAPA-TDNN) as our primary model. We also incorporate one-class learning with channel-robust training strategies to further learn a channel-invariant speech representation. Our submission achieved EER 20.33% in the DF task; EER 5.46% and min-tDCF 0.3094 in the LA task.

**Index Terms:** synthetic speech detection, anti-spoofing, deepfake, data augmentation, channel robustness, ASVspoof 2021

## 1. Introduction

Automatic speaker verification (ASV) systems have been widely used in biometric authentication, where a person's identity is automatically verified with acceptance or rejection by analyzing speech utterances [1, 2]. However, past research has shown that the ASV system is vulnerable to spoofing attacks, where attackers pretend to be the target speaker by presenting false but similar-to-bona-fide speech trials [3]. Anti-spoofing has been a research topic to deal with this issue.

Synthetic speech detection (SSD) aims to distinguish the text-to-speech (TTS) and voice conversion (VC) attacks from bona fide speech. It is an important issue in anti-spoofing since the speech synthesis techniques are fast developing and will cause more threats to speaker verification systems [4, 5]. In a recent study [6], some synthetic speech is even perceptually non-distinguishable from bona fide speech.

One issue in SSD is the generalization ability towards unseen spoofing attacks during training. Efforts have been put into the study of generalization with a focus on feature engineering [7], backend model [8], loss function [9, 10], data augmentation [11], and model ensemble [12]. According to the result of the ASVspoof 2019 Challenge, for the LA scenario, very few systems deliver EERs below 5% [13], which suggests the necessity of better countermeasures for synthetic speech attacks.

\* The first three authors have equal contributions, presented in alphabetical order.

Besides, there is a big performance gap between training and evaluation, suggesting generalization issues to unseen attacks.

Another issue is the robustness to the variability of recording conditions, including background noise and channel effects. The performance degradation of state-of-the-art spoofing detectors under unseen conditions has been revealed in some cross-dataset studies for SSD [14, 15, 16, 17]. Of all the factors presenting cross-dataset effects, we tested and verified the channel effect as a reason for performance degradation of SSD systems in [17]. Without considering these effects, the systems may fail to generate robust results under different variations.

The logical access (LA) track of the ASVspoof 2021 Challenge [18] also aims at designing generalized countermeasures that are robust to codec and transmission channel variability. In addition, a new SSD task called speech deepfake (DF) is proposed in ASVspoof 2021 to detect fake audio without ASV systems. For this task, developing systems that can be applied in multiple and unknown conditions such as audio compression is also critical. Furthermore, both LA and DF use attacks unseen during training for evaluation, as in ASVspoof 2019. This makes the ASVspoof 2021 challenge a great test bench for evaluating the generalization ability and channel robustness of SSD systems.

In this paper, we develop a system based on our previous work for addressing the generalization ability with one-class learning [9] and the robustness issue with channel-robust strategies [17]. Our main contributions are threefold:

- We design several data augmentation methods depending on the task focus: for the DF task, we applied Fraunhofer MPEG layer III (MP3) and advanced audio codec (AAC) audio compression codecs; for the LA task, we introduce three common transmission codec degradation including landline, cellular, and Voice over Internet Protocol (VoIP). We also propose to involve device impulse responses to the above two augmentation pipelines as an additional step to remove the dataset bias introduced by ASVspoof 2019 training dataset.
- We employ a state-of-the-art backbone in several speaker recognition tasks, ECAPA-TDNN [19] as the primary architecture, and we ensemble its variants to further improve the final performance. To the best of our knowledge, we are the first to have ECAPA-TDNN in the ASV anti-spoofing task.
- We adapt the channel-robust training strategies proposed in [17] to pair with the new data augmentation methods.

## 2. System Description

### 2.1. Data Augmentation

In our previous work [17], we verified that channel effects are an important reason for cross-dataset performance degradation,

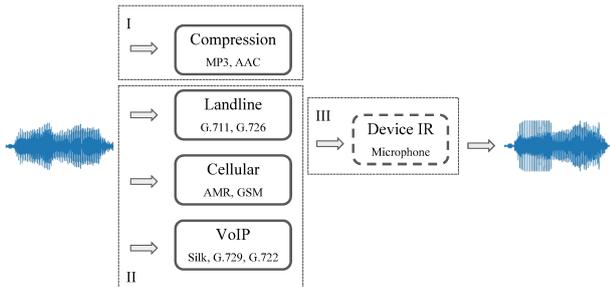


Figure 1: Illustration of our proposed data augmentation pipeline. The first step is codec augmentation, where part I is for the DF task and part II is for the LA task. The second step is device augmentation, which is optional for both tasks.

and data augmentation methods are shown effective to improve the cross-dataset performance of anti-spoofing systems. To address the compression variability in the DF task and the transmission variability in the LA task, we design different augmentation methods correspondingly.

Figure 1 shows the pipeline for our proposed data augmentation methods. We utilize four groups of different lossy codec and several impulse responses collected in the open-source acoustic simulator [20]. Different from the previous work where we only explored the device IRs, we first apply codec degradation, i.e., compression and transmission, and then optionally convolve them with recording device IRs.

### 2.1.1. Codec Augmentation

**Compression Augmentation for DF.** Compression refers to compression-decompression algorithms that are used to process the audio files, the deepfake speech may be spread across multiple conditions such as general audio compression rather than telephony. As shown in Figure 1 part I, for our compression augmentation, we apply MP3 and AAC codecs in the acoustic simulator.

**Transmission Augmentation for LA.** Transmission refers to codecs that the speech data is transmitted across before being processed by anti-spoofing systems. The transmission process would introduce degradation due to different bandwidths packet loss and sampling frequencies. However, the degradation should not be considered as artifacts that we aim to detect for discriminating spoofing attacks. As shown in Figure 1 part II, for transmission degradation, we apply landline, cellular and VoIP codec groups, which are the most commonly used transmission channels. To be specific, the landline codec includes  $\mu$ -law and A-law companding and adaptive-differential PCM. Cellular includes GSM and narrow-band and wide-band advance multi-rate (AMR-NB and AMR-WB). And VoIP includes ITU G.722, ITU G.729 standards besides SILK and SILK-WB. All of the above-mentioned codecs are equipped with different bitrate options. After going through these different transmission codecs, we resample all of the resulting audio files with 16 kHz for further processing.

### 2.1.2. Device Augmentation

As shown in our previous study [17] where we only applied device augmentation, it is beneficial to introduce various device impulse responses to mitigate the device bias induced by the ASVspoof2019LA training dataset and work robustly across

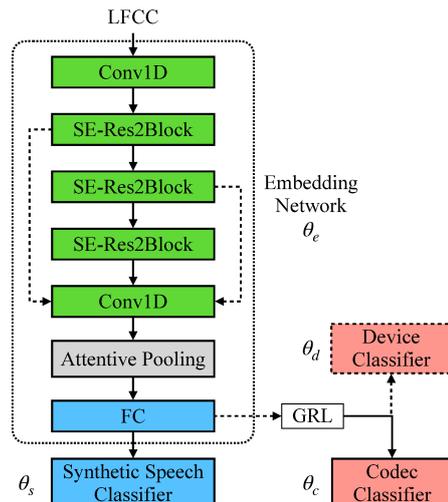


Figure 2: Proposed model architecture with a channel-robust training strategy. Dashed arrows represent optional operations.

channel variations. As shown in Figure 1 part III, we propose to convolve the original audio data with recording device IRs after applying compression or transmission augmentation.

## 2.2. Model Architecture: ECAPA-TDNN

Time-delay neural networks (TDNN) [21, 22] were originally proposed in modeling long-range temporal dependencies and have long been successfully used in speech and speaker recognition tasks [23, 24]. It is similar to the 1-dimensional dilated convolutional neural network (CNN) where only some specific frames in the convolution context window are used. Compared to ordinary CNN, it requires fewer parameters and achieves better results when processing the same width of temporal context [22, 25].

In this paper, we employ an improved version of TDNN called ECAPA-TDNN [19], which has introduced several enhancements to improve the original TDNN including Squeeze-Excitation (SE) blocks, multi-scale Res2Net features, multi-layer feature aggregation, and channel-dependent attentive statistics pooling. The model architecture is illustrated in Figure 2. Specifically, the 1-dimensional SE-Res2Net layer first utilizes SE blocks to re-scale the channels of feature maps with global context information in the convolutional blocks. The Res2Net architecture further improves the performance and simultaneously reduces the total number of parameters through grouped convolutions, this layer is similar to a recent successful anti-spoofing system [26]. Another key component in ECAPA-TDNN is the aggregation of multi-level features through skip connections, as illustrated with the dashed lines between the green blocks in Figure 2.

With the combination of the above-mentioned modules, the ECAPA-TDNN architecture recently achieved the best performance in all three tracks of speaker verification in VOXSRC [27] and the text-independent task of SdSV Challenge on short-duration speaker verification [28]. As the SSD task also aims to learn a discriminative speaker embedding to distinguish spoofing attacks and considers temporal contextual modeling, we believe that the ECAPA-TDNN architecture is also a good choice for the deep learning backbone.

In this paper, we studied and compared six setups of ECAPA-TDNN, shown in Table 1, by varying the number of filters of convolutional layers, and whether or not to insert the multi-layer feature aggregation (MFA) block and channel- and context-dependent statistics pooling (CCSP). Although the channel- and context-dependent vector before pooling only marginally improves speaker recognition performance in the original ECAPA-TDNN system [19], we include this operation as an option here as it could be beneficial in our application.

Table 1: ECAPA-TDNN backbone variants

System	# of Filters	CCSP	MFA
ECAPA-1	512	✓	✗
ECAPA-2		✗	✓
ECAPA-3		✓	✓
ECAPA-4	1024	✓	✗
ECAPA-5		✗	✓
ECAPA-6		✓	✓

### 2.3. Training Strategy

To improve the generalization ability, the idea of one-class learning [9] with OC-Softmax as the loss function is adopted, where two different margins are introduced for compacting the bona fide speech in the embedding space and isolating the spoofing attacks, to prevent overfitting known conditions.

To improve the channel robustness, we adopt the channel-robust training strategies proposed in [17] to take advantage of the augmented data. As the new evaluation dataset in ASVspoof 2021 introduces new channel variability, we assume that it has different distributions from the training data.

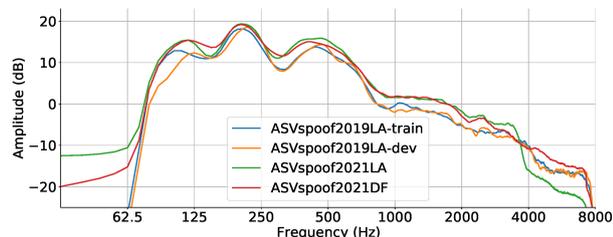


Figure 3: Comparison of average magnitude spectra of utterances across different sets of data.

In Figure 3, we compare the average magnitude spectrum of the LA and DF datasets of the ASVspoof 2021 challenge with the ASVspoof 2019 training data across all utterances of each dataset to illustrate the channel effects. We can see that the average spectra are very different for LA, especially below 62.5 Hz and above 4 kHz. The low-frequency component could be caused by some channel noise. And the high-frequency component could be missing for some utterances, due to the fact that some transmission codecs, such as cellular or landline sampled at 8kHz, will lose the spectral information above Nyquist frequency. For DF, the main difference exists in frequencies below 62.5 Hz, which could also be caused by channel noise. With this observation, we verify that channel mismatch does exist across the training and evaluation data for this challenge. As such, with this observation, we assume that the channel-robust strategies would improve the performance.

We follow the augmentation structure in [17] and use plain augmentation (AUG) training and adversarial (ADV-AUG) training strategies. As illustrated in Figure 2, in the AUG strategy, only the embedding network, parameterized by  $\theta_e$ , and the synthetic speech classifier parameterized by  $\theta_s$  are utilized. In ADV-AUG strategy, to fit the data augmentation methods in Section 2.1, we design two additional different classifiers, parameterized by  $\theta_c$  and  $\theta_d$  respectively, to deal with two categories of augmentation, codec and device. The overall training objective is:

$$(\hat{\theta}_e, \hat{\theta}_s) = \arg \min_{\theta_e, \theta_s} \mathcal{L}_s(\theta_e, \theta_s) - \lambda_1 \mathcal{L}_c(\theta_e, \hat{\theta}_c) - \lambda_2 \mathcal{L}_d(\theta_e, \hat{\theta}_d),$$

$$(\hat{\theta}_c) = \arg \min_{\theta_c} \mathcal{L}_c(\hat{\theta}_e, \theta_c),$$

$$(\hat{\theta}_d) = \arg \min_{\theta_d} \mathcal{L}_d(\hat{\theta}_e, \theta_d).$$

## 3. Experimental Setup

### 3.1. Dataset

**Training Data.** According to the rule of the challenge, we are only allowed to use the training and development partitions of the ASVspoof 2019 database [29] to develop our model. No external speech data is allowed, but non-speech data such as impulse responses are allowed. For both LA and DF tasks, we use 2019train as the source data and perform data augmentation with methods shown in Section 2.1.

For each of the augmentation methods mentioned in Section 2.1.1, codec parameters such as bit rate, packet loss are randomly sampled to handle different settings. This results in 60 transmission codec options for LA and 24 compression codec options for DF. For transmission only and device-transmission paired degradation pipelines, we randomly sampled and created 21 paralleled transmission codec audio files and 6 paralleled compression codec audio files for LA and DF, respectively. For the additional device augmentation step, we sampled 12 device impulse responses including iPhone, iPad, telephone, and several different microphones.

**Evaluation Data.** The challenge uses a new set of evaluation data for each task. For the DF task, the data was primarily generated by ASVspoof2019 LA eval data going through several unknown compression codec channels, which result in a total of 611,829 trials. As for the LA task, the TTS, VC and hybrid spoofing attacks were the same as those from the ASVspoof 2019 LA evaluation partition. Similarly, it was degraded by different unknown transmission channels, totaling 181,566 trials.

### 3.2. Feature Extraction

Following our previous works [9, 17], we extract 60-dimensional linear frequency cepstral coefficients (LFCCs) with a 20ms frame size and 10ms hop size. To form batches, we set a fixed length as exactly 750 frames. For shorter utterances, we apply repeat padding until the utterance is longer than the fixed length. Then we randomly truncate along the time axis to choose a consecutive sequence.

### 3.3. Training Details

After feature extraction from augmented data, we train several sub-systems with Adam optimizer to update the weights in models. Each training process is run on a single NVIDIA GTX 1080 Ti GPU with a learning rate of 0.0005, over 200 epochs,

Table 2: Details and results of our submitted UR-AIR systems for DF and LA tasks. Both systems are fused by several sub-systems.

Task	Index	Feature	Backbone	Loss	Strategy	Augmentation		Results	
						Codec	Device	EER	min-tDCF
DF	1		ECAPA-3		AUG	✓	✗	20.33	N/A
	2		ECAPA-3		AUG	✓	✓		
	3	LFCC	ECAPA-3	OC-Softmax	ADV-AUG	✓	✓		
	4		ECAPA-3		ADV-AUG	✓	✗		
	5		ResNet		ADV-AUG	✓	✓		
LA	1		ECAPA-2	OC-Softmax	AUG	✓	✗	5.46	0.3094
	2		ECAPA-1	OC-Softmax	AUG	✓	✗		
	3		ECAPA-1	P2SGrad	AUG	✓	✗		
	4	LFCC	ECAPA-3	OC-Softmax	ADV-AUG	✓	✗		
	5		ECAPA-3	OC-Softmax	AUG	✓	✗		
	6		ECAPA-3	OC-Softmax	AUG	✓	✓		
	7		ECAPA-3	OC-Softmax	ADV-AUG	✓	✓		

and a batch size of 64. The learning rate is reduced by half every 30 epochs. We set  $\lambda_1$  and  $\lambda_2$  in the ADV-AUG training strategy as 0.05. Then we select the model with the lowest validation loss for evaluation. The implementation is available at [github.com/yzyouzhang/ASVspooof2021\\_AIR](https://github.com/yzyouzhang/ASVspooof2021_AIR).

### 3.4. Fused Systems

We perform the logistic regression with Bosaris [30] toolkit to fuse several sub-systems. We choose several ECAPA-TDNN based systems for the fusion stage. Besides the primary models we described in Section 2.2, we also fuse a secondary ResNet system shown in Table 2. In the LA task, we also include a system with P2SGrad loss function proposed in [10], which achieves 4.77% EER and min-tDCF 0.2545 in the progress phase<sup>1</sup>, although it is not stable in other settings. The final fused system is presented in Table 2.

### 3.5. Evaluation Metrics

The output score of the SSD system indicates the confidence of the utterance belonging to the bona fide speech. Equal Error Rate (EER) is calculated by setting a threshold on the output score such that the miss probability is equal to the false alarm probability. The lower EER denotes the better discrimination ability of the SSD system. Tandem detection cost function (t-DCF) [31] assesses the influence of anti-spoofing systems on the reliability of an ASV system. To compare different anti-spoofing systems, the ASV system is fixed in the challenge. The lower t-DCF indicates the better reliability of the SSD system on ASV. The t-DCF is slightly revised compared to the ASVspooof 2019 challenge.

Based on the evaluation plan of the ASVspooof2021 challenge, the metric for the DF task is EER. For the LA task, the main metric is t-DCF, but we also include EER in our analysis.

## 4. Results and Analysis

### 4.1. Performance Analysis of Our Submitted System

The overall performance in the evaluation phase is shown in Table 2. The DF detailed results are reported in Figure 4 with

<sup>1</sup>In the progress phase, only part of the data is used to calculate the metrics. The ground truth label is withheld in both the progress phase and the evaluation phase.

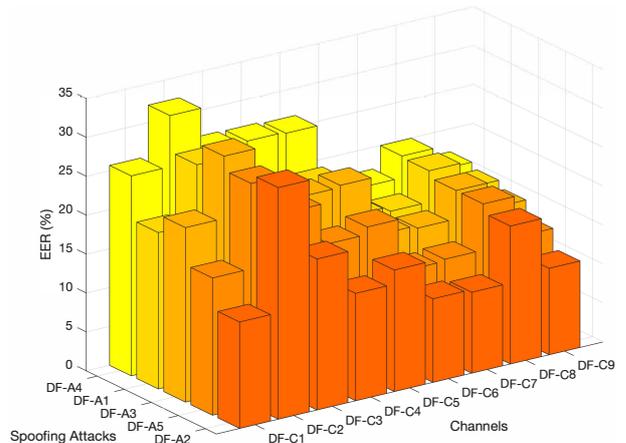


Figure 4: Detailed performance of DF submission on different channels and attacks. The spoofing attacks are traditional vocoder (A1), waveform concatenation (A2), autoregressive neural vocoder (A3), non-autoregressive neural vocoder (A4), unknown (A5). C1-C9 represents different compression codecs.

different channels and attacks. Among different channels, we observe that DF-C2 (low-quality MP3) is the most difficult for our system. While from the perspective of spoofing attacks, non-autoregressive neural vocoder is the hardest one to detect.

In Softmax loss-based systems for verification tasks, it has been found that the learned embeddings tend to correlate their norms with the quality of input features [32]: the lower the input quality is, the smaller its embedding norm is. In this paper, we also observe the same problem in the DF task. Figure 5 shows an example norm histogram of the feature vectors of the training set and evaluation set of speech utterances with different compression codecs. We can observe a significant distribution mismatch between the two datasets, which can possibly explain this poor performance in the DF task.

As for LA, the detailed results are reported in Figure 6 based on different waveform generation methods: vocoder (A07, A09, A14, A18), neural waveform (A08, A10, A12, A15), waveform filtering (A13, A17), spectral filtering (A19) and waveform concatenation (A13, A16) [33]. In accordance

Table 3: Comparison of EER (%) performance pooled over attacks on DF.

System	DF-C1	DF-C2	DF-C3	DF-C4	DF-C5	DF-C6	DF-C7	DF-C8	DF-C9	Pooled
CQCC-GMM	19.48	48.86	20.37	<b>19.55</b>	<b>20.27</b>	17.92	<b>14.42</b>	49.41	17.39	25.56
LFCC-GMM	<b>17.39</b>	39.20	<b>17.97</b>	20.95	21.43	22.16	14.83	39.03	26.65	25.25
LFCC-LCNN	23.19	34.21	23.88	25.22	23.85	19.06	17.10	28.35	18.54	23.48
RawNet2	26.98	<b>27.63</b>	27.49	26.72	27.23	18.80	18.67	<b>18.74</b>	19.10	22.38
<b>Ours</b>	22.04	29.56	24.76	22.86	23.04	<b>16.04</b>	15.74	19.54	<b>16.17</b>	<b>20.33</b>

Table 4: Comparison of EER (%) performance pooled over attacks on LA.

System	LA-C1	LA-C2	LA-C3	LA-C4	LA-C5	LA-C6	LA-C7	Pooled
CQCC-GMM	10.57	14.76	20.58	11.61	13.58	14.01	11.21	15.62
LFCC-GMM	12.72	21.21	35.55	15.28	18.76	18.46	12.73	19.30
LFCC-LCNN	6.71	8.89	12.02	6.34	9.25	11.00	6.66	9.26
RawNet2	5.84	6.59	16.72	6.41	6.33	10.66	7.95	9.50
<b>Ours</b>	<b>4.73</b>	<b>4.47</b>	<b>6.20</b>	<b>5.06</b>	<b>4.71</b>	<b>5.18</b>	<b>5.82</b>	<b>5.46</b>

Table 5: Comparison of min-tDCF performance pooled over attacks on LA.

System	LA-C1	LA-C2	LA-C3	LA-C4	LA-C5	LA-C6	LA-C7	Pooled
CQCC-GMM	0.2858	0.5116	0.6183	0.3715	0.4703	0.4791	0.2935	0.4974
LFCC-GMM	0.3615	0.6334	0.8307	0.4409	0.5819	0.6061	0.3661	0.5758
LFCC-LCNN	0.2186	0.3354	0.4286	0.2218	0.3505	0.4217	<b>0.2166</b>	0.3445
RawNet2	0.2073	0.3584	0.6367	0.2387	0.3489	0.4959	0.2908	0.4257
<b>Ours</b>	<b>0.1902</b>	<b>0.2911</b>	<b>0.3720</b>	<b>0.2104</b>	<b>0.2996</b>	<b>0.3618</b>	0.2186	<b>0.3094</b>

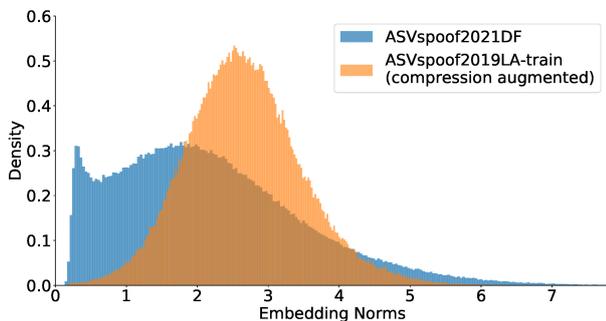


Figure 5: Density histogram of the calculated norms of the feature embeddings of one system across the training and evaluation data for DF.

with the analysis of the generalization performance in the ASVspoof2019 database [13, 29] where A17 was considered as the worst-case scenario, our system still could not improve upon this spoofing algorithm. Besides, we also find it difficult for our system to detect A18, especially transmitted through LA-C3 (public switched telephone network (PSTN)). It could be attributed to that PSTN is not included in our transmission codec augmentation. We believe that our augmentation with landline codec simulates only part of the PSTN process. However, even for LA-C1, i.e., no codec, our system shows a higher EER compared to our previous work [9]. This puzzles us and further investigations are needed after we have access to the complete information of the test data.

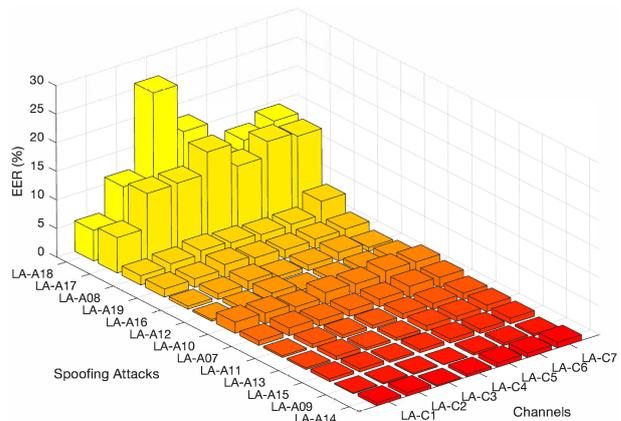


Figure 6: Detailed performance of LA submission on different channels and attacks. A07-A19 represents different spoofing attacks in the evaluation data, as defined in the ASVspoof2019 database. C1-C7 represents different codec effects.

#### 4.2. Comparison with Baseline Systems

We compare our submitted system with the four baseline systems LFCC-GMM [7], CQCC-GMM [34], LFCC-LCNN [10], RawNet2 [35], provided by the organizers. Table 3 and Table 4 present the EER performance of all baseline systems and our proposed system for DF and LA respectively. For the DF task, we achieve better overall performance. In individual DF channels, it achieves comparable results on DF-C2, DF-C7, and DF-

C8, and marginally improves baseline systems on DF-C6 and DF-C9. For the LA task, our proposed system performs better than all of the baseline systems and is more robust, resulting in evenly distributed low EERs on all spoofing conditions. We also present min-tDCF performance for LA in Table 5.

### 4.3. Model Comparison

Table 6 compares different backbone models in the progress phase. By comparing the theoretical amount of multiply-accumulate (MAC) operations among every sub-system and the number of parameters, we can easily find that LCNN has both the least MACs and parameters while ECAPA-4 and ECAPA-6 have the largest. Although more GPU capacity and time are needed for training ECAPA models, they demonstrated lower EERs as well as min-tDCF compared to either LCNN or Res2Net. By comparing a small-scale ECAPA-TDNN (C=512) with a large-scale version (C=1024), we observe that the smaller versions outperform the larger ones which have over twice of parameters and MACs, indicating that large models have already over-fitted on the training data. By comparing systems ECAPA-1 and ECAPA-2 with ECAPA-3, we can see that the improvement of context vector before pooling is as significant as MFA in anti-spoofing task in terms of min-tDCF.

Table 6: Comparison of different backbone models on the ASVspoof 2021 LA task in the progress phase.

System	Parameters	MACs	EER	min-tDCF
LCNN	0.52 M	0.63 G	7.28	0.2738
Res2Net	0.89 M	1.47 G	7.28	0.2801
ECAPA-1	6.34 M	3.87 G	5.31	0.2732
ECAPA-2	5.94 M	3.57 G	5.14	0.2759
ECAPA-3	6.34 M	3.87 G	5.13	0.2626
ECAPA-4	14.75 M	9.89 G	5.77	0.2725
ECAPA-5	14.36 M	9.59 G	6.39	0.2818
ECAPA-6	14.75 M	9.89 G	6.93	0.2942

### 4.4. Effect of Model Fusion

In Figure 7, we plot the detection error tradeoff (DET) curve to show the miss probability and false alarm probability trade-off by varying the threshold for the output synthetic speech classification score. We compute a DET curve for each individual system on the codec-augmented ASVspoof2019LA-dev dataset during the progress phase. From Figure 7, it is obvious that the fused model strongly outperforms individual ones in both DF and LA tasks. Hence, we submit our fused system during the evaluation phase. Note that the conclusion that model fusion is better than individuals is drawn from the performance on the augmented development set. We believe further investigations are needed on the ASVspoof 2021 evaluation set during the post-evaluation phase.

## 5. Conclusions

In this paper, we introduced UR-AIR system for the channel-robust anti-spoofing task. Specifically, to improve the robustness over channel variability, we use an acoustic simulator to apply transmission codec, compression codec, and convolutional device impulse responses to the training data. To learn a reliable embedding from the augmented dataset, we use sev-

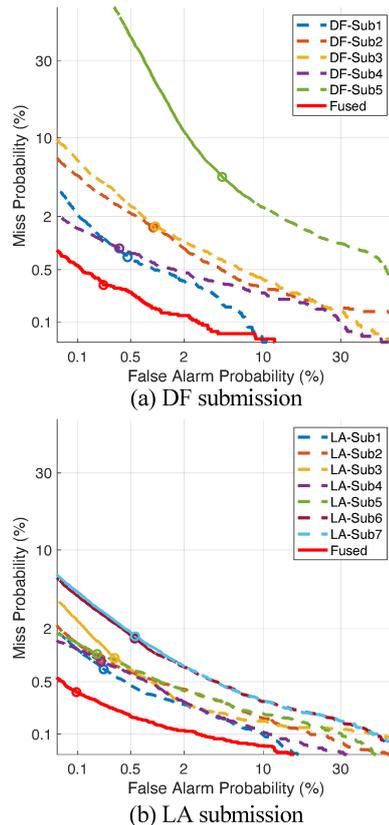


Figure 7: Performance of model fusion of our submissions on the augmented development set. Dashed curves represent the DET curve of individual systems and the red solid curve represents that of our fused system. The circles on the DET curves represent the operating points of the min-tDCF. The indices of the sub-systems are consistent with those in Table 2.

eral variants of ECAPA-TDNN as our backbone neural network architecture. We also incorporate one-class learning with channel-robust training strategies to further learn a channel-invariant speech representation. Our submission achieved considerable improvement over baseline systems in both DF and LA tasks of the ASVspoof 2021 challenge.

## 6. Acknowledgments

This work was supported by National Science Foundation grant No. 1741472 and funding from Voice Biometrics Group. You Zhang would like to thank the synergistic activities provided by the NRT program on AR/VR funded by NSF grant DGE-1922591. The authors would also like to thank the organizers for providing additional results breakdown for our submissions.

## 7. References

- [1] Judith A Markowitz, “Voice biometrics,” *Communications of the ACM*, vol. 43, no. 9, pp. 66–73, 2000.
- [2] John HL Hansen and Taufiq Hasan, “Speaker recognition by machines and humans: A tutorial review,” *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74–99, 2015.
- [3] Zhizheng Wu, Nicholas Evans, Tomi Kinnunen, Junichi Yamagishi, Federico Alegre, and Haizhou Li, “Spoofing

- and countermeasures for speaker verification: A survey,” *Speech Communication*, vol. 66, pp. 130–153, 2015.
- [4] Zhizheng Wu, Phillip L. De Leon, Cenk Demiroglu, Ali Khodabakhsh, Simon King, Zhen-Hua Ling, Daisuke Saito, Bryan Stewart, Tomoki Toda, Mirjam Wester, and Junichi Yamagishi, “Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 768–783, 2016.
  - [5] Madhu R Kamble, Hardik B Sailor, Hemant A Patil, and Haizhou Li, “Advances in anti-spoofing: from the perspective of ASVspoof challenges,” *APSIPA Transactions on Signal and Information Processing*, vol. 9, 2020.
  - [6] Nicolas M. Müller, Karla Markert, and Konstantin Böttinger, “Human perception of audio deepfakes,” *arXiv preprint arXiv:2107.09667*, 2021.
  - [7] Md Sahidullah, Tomi Kinnunen, and Cemal Hanilçi, “A comparison of features for synthetic speech detection,” in *Proc. Interspeech*, 2015, pp. 2087–2091.
  - [8] Joao Monteiro, Jahangir Alam, and Tiago H Falk, “Generalized end-to-end detection of spoofing attacks to automatic speaker recognizers,” *Computer Speech & Language*, vol. 63, pp. 101096, 2020.
  - [9] You Zhang, Fei Jiang, and Zhiyao Duan, “One-class learning towards synthetic voice spoofing detection,” *IEEE Signal Processing Letters*, vol. 28, pp. 937–941, 2021.
  - [10] Xin Wang and Junich Yamagishi, “A comparative study on recent neural spoofing countermeasures for synthetic speech detection,” *Proc. Interspeech*, 2021.
  - [11] Tianxiang Chen, Avrosh Kumar, Parav Nagarsheth, Ganesh Sivaraman, and Elie Khoury, “Generalization of audio deepfake detection,” in *Proc. Odyssey*, 2020, pp. 132–137.
  - [12] João Monteiro, Jahangir Alam, and Tiago H Falk, “An ensemble based approach for generalized detection of spoofing attacks to automatic speaker recognizers,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6599–6603.
  - [13] Massimiliano Todisco, Xin Wang, Ville Vestman, Md. Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi H. Kinnunen, and Kong Aik Lee, “ASVspoof 2019: Future horizons in spoofed and fake audio detection,” in *Proc. Interspeech*, 2019, pp. 1008–1012.
  - [14] Rohan Kumar Das, Jichen Yang, and Haizhou Li, “Assessing the scope of generalized countermeasures for anti-spoofing,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6589–6593.
  - [15] Rohan Kumar Das, Tomi Kinnunen, Wen-Chin Huang, Zhen-Hua Ling, Junichi Yamagishi, Zhao Yi, Xiaohai Tian, and Tomoki Toda, “Predictions of subjective ratings and spoofing assessments of voice conversion challenge 2020 submissions,” in *Proc. Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge*, 2020, pp. 99–120.
  - [16] Cemal Hanilci, Tomi Kinnunen, Md Sahidullah, and Aleksandr Sizov, “Spoofing detection goes noisy: An analysis of synthetic speech detection in the presence of additive noise,” *Speech Communication*, vol. 85, pp. 83–97, 2016.
  - [17] You Zhang, Ge Zhu, Fei Jiang, and Zhiyao Duan, “An empirical study on channel effects for synthetic voice spoofing countermeasure systems,” *Proc. Interspeech*, 2021.
  - [18] Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, and Hector Delgado, “ASVspoof2021: accelerating progress in spoofed and deep fake speech detection,” in *Proc. ASVspoof 2021 Workshop*, 2021.
  - [19] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, “ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification,” *Proc. Interspeech*, pp. 3830–3834, 2020.
  - [20] Marc Ferras, Srikanth Madikeri, Petr Motlicek, Subhadeep Dey, and Hervé Bourlard, “A large-scale open-source acoustic simulator for speaker recognition,” *IEEE Signal Processing Letters*, vol. 23, no. 4, pp. 527–531, 2016.
  - [21] Alex Waibel, Toshiyuki Hanazawa, Geoffrey Hinton, Kiyohiro Shikano, and Kevin J Lang, “Phoneme recognition using time-delay neural networks,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, 1989.
  - [22] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, “A time delay neural network architecture for efficient modeling of long temporal contexts,” in *Proc. Interspeech*, 2015.
  - [23] Kyu J Han, Jing Pan, Venkata Krishna Naveen Tadala, Tao Ma, and Dan Povey, “Multistream cnn for robust acoustic modeling,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6873–6877.
  - [24] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.
  - [25] Hossein Zeinali, Lukáš Burget, Johan Rohdin, Theodoros Stafylakis, and Jan Cernocky, “How to improve your speaker embeddings extractor in generic toolkits,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6141–6145.
  - [26] Xu Li, Na Li, Chao Weng, Xunying Liu, Dan Su, Dong Yu, and Helen Meng, “Replay and synthetic speech detection with res2net architecture,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6354–6358.
  - [27] Arsha Nagrani, Joon Son Chung, Jaesung Huh, Andrew Brown, Ernesto Coto, Weidi Xie, Mitchell McLaren, Douglas A Reynolds, and Andrew Zisserman, “Voxsrc 2020: The second voxceleb speaker recognition challenge,” *arXiv preprint arXiv:2012.06867*, 2020.
  - [28] Hossein Zeinali, Kong Aik Lee, Jahangir Alam, and Lukas Burget, “SdSV challenge 2020: Large-scale evaluation of short-duration speaker verification,” in *Proc. Interspeech*, 2020, pp. 731–735.

- [29] Xin Wang, Junichi Yamagishi, Massimiliano Todisco, Héctor Delgado, Andreas Nautsch, Nicholas Evans, Md Sahidullah, Ville Vestman, Tomi Kinnunen, Kong Aik Lee, et al., “ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech.” *Computer Speech & Language*, vol. 64, pp. 101114, 2020.
- [30] Niko Brümmer and Edward De Villiers, “The bosaris toolkit: Theory, algorithms and code for surviving the new dcf,” *arXiv preprint arXiv:1304.2865*, 2013.
- [31] Tomi Kinnunen, Héctor Delgado, Nicholas Evans, Kong Aik Lee, Ville Vestman, Andreas Nautsch, Massimiliano Todisco, Xin Wang, Md Sahidullah, Junichi Yamagishi, et al., “Tandem assessment of spoofing countermeasures and automatic speaker verification: Fundamentals.” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2195–2210, 2020.
- [32] Rajeev Ranjan, Carlos D Castillo, and Rama Chellappa, “L2-constrained softmax loss for discriminative face verification,” *arXiv preprint arXiv:1703.09507*, 2017.
- [33] Hossein Zeinali, Themis Stafylakis, Georgia Athanopoulou, Johan Rohdin, Ioannis Gkinis, Lukas Burget, and Jan Cernocky, “Detecting spoofing attacks using vgg and sincnet: BUT-Omilia submission to ASVspoof 2019 challenge,” *Proc. Interspeech*, pp. 1073–1077, 2019.
- [34] Massimiliano Todisco, Héctor Delgado, and Nicholas WD Evans, “A new feature for automatic speaker verification anti-spoofing: Constant q cepstral coefficients.” in *Proc. Odyssey*, 2016, vol. 2016, pp. 283–290.
- [35] Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher, “End-to-end anti-spoofing with RawNet2,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6369–6373.