

ASVspoof 5: crowdsourced speech data, deepfakes, and adversarial attacks at scale

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-Weon Jung, Hye-Jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi Kinnunen, et al.

▶ **To cite this version:**

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-Weon Jung, Hye-Jin Shim, et al.. ASVspoof 5: crowdsourced speech data, deepfakes, and adversarial attacks at scale. The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024), ISCA, Aug 2024, Kos Island, Greece. pp.1-8, ⟨10.21437/ASVspoof.2024-1⟩. ⟨hal-04803294⟩

HAL Id: hal-04803294

https://hal.science/hal-04803294v1

Submitted on 25 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)
31 August 2024, Kos, Greece

ASVspoof 5: Crowdsourced Speech Data, Deepfakes, and Adversarial Attacks at Scale

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-weon Jung, Hye-jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi Kinnunen, Nicholas Evans, Kong Aik Lee, Junichi Yamagishi

ASVspoof consortium

http://www.asvspoof.org/

Abstract

ASVspoof 5 is the fifth edition in a series of challenges which promote the study of speech spoofing and deepfake attacks, and the design of detection solutions. Compared to previous challenges, the ASVspoof 5 database is built from crowdsourced data collected from a vastly greater number of speakers in diverse acoustic conditions. Attacks, also crowdsourced, are generated and tested using surrogate detection models, while adversarial attacks are incorporated for the first time. New metrics support the evaluation of spoofing-robust automatic speaker verification (SASV) as well as stand-alone detection solutions, i.e., countermeasures without ASV. We describe the two challenge tracks, the new database, the evaluation metrics, baselines, and the evaluation platform, and present a summary of the results. Attacks significantly compromise the baseline systems, while submissions bring substantial improvements.

1. Introduction

The ASVspoof initiative was conceived to foster progress in the development of detection solutions, also referred to as countermeasures (CMs) and presentation attack detection (PAD) solutions, to discriminate between bona fide and spoofed or deepfake speech utterances. ASVspoof 5 is the fifth edition in a series of previously-biennial challenges [1–4] and has evolved in terms of evaluation tracks, the database and spoofing attacks, and evaluation metrics.

While the 2021 challenge edition involved distinct logical access (LA), physical access (PA), and speech deepfake (DF) sub-tasks [5], ASVspoof 5 takes the form of a single, combined LA and DF task, but encompasses two tracks: (i) standalone spoofing and speech deepfake detection (CM, no ASV) and (ii) spoofing-robust automatic speaker verification (SASV). Track 1 is similar to the DF track of the previous 2021 challenge. It reflects a scenario in which an attacker has access to the voice data of a targeted victim, e.g. data posted to social media. The attacker is assumed to use public data and speech deepfake technology to generate spoofed speech resembling the voice of the victim and then, e.g., to re-post generated recordings to social media to defame the victim. Speech data, both bona fide and spoofed, may be compressed using conventional codecs (e.g., mp3) or contemporary neural codecs.

Track 2 shares the same goal as the LA sub-task of previous ASVspoof editions and the SASV 2022 Challenge [6]. Track 2 assumes a telephony scenario where synthetic and converted speech are injected into a communication system (e.g., a telephone line) without any acoustic propagation. Participants can elect to develop single classifiers or separate, fused ASV and CM sub-systems. They can use either a pre-trained ASV sub-system provided by the organisers or can optimise their own bespoke system.

Participants are furthermore provided with an entirely new ASVspoof 5 database. Source data and attacks, both crowdsourced, encompass greater acoustic variation than earlier ASVspoof databases. The objective is to evaluate the threat of spoofing and deepfake attacks forged using non-studio-quality data and optimised to compromise not just ASV subsystems but also CM sub-systems. Source data, collected from a vastly greater number of speakers than for earlier ASVspoof databases, is extracted from the Multilingual Librispeech (MLS) English partition [7]. In addition to the use of new spoofing attacks implemented using the latest text-to-speech (TTS) synthesis and voice conversion (VC) algorithms, adversarial attacks are introduced for the first time and combined with spoofing attacks.

Also new is an *open* condition for both Tracks 1 and 2. In contrast to the traditional *closed* condition, for which participants are restricted to use the specified data protocol, for the open condition participants have the opportunity to use external data and pre-trained speech foundation models, subject to there being no overlap between training data (i.e. that used for training foundation models) and challenge evaluation data.

A new suite of evaluation metrics is also introduced. Inspired by the NIST SREs [8], we adopt the minimum detection cost function (minDCF) as the primary metric for Track 1. The log-likelihood-ratio cost function (C_{llr}) and actual DCF are also used to gauge not only discrimination but also calibration performance. The recently proposed architecture-agnostic DCF (a-DCF) [9] is used as the primary metric for Track 2, with the tandem detection cost function (t-DCF) [10] and tandem equal error rate (t-EER) [11] being complementary.

We present an overview of the database, the two challenge tracks, common systems and baselines, and the evaluation metrics. Spoofing and deepfake attacks and their performance in fooling an ASV system are also described. Finally, we report a summary of system performance for the baselines and those submitted by 54 challenge participants.

2. Database

The new ASVspoof 5 database has evolved in two aspects: source data and attack algorithms. To evaluate the performance of CM and SASV systems in detecting spoofing attacks forged using non-studio-quality data, the new database is constructed using data sourced from the MLS English dataset [7]. The latter incorporates data from more than 4k speakers, recorded

10.21437/ASVspoof.2024-1

Table 1: Summary of ASVspoof 5 database key statistics. The number of target speakers is listed in braces. Target speakers relate to Track 2 (T2) only and are not defined for Track 1 (T1). Figures do not include enrollment utterances.

	#. speaker		#. utterances		#. spf.
	Female	Male	Bona fide	Spoofed	attack
Trn.	196	204	18,797	163,560	8
Dev.	392 (196)	393 (202)	31,334	109,616	8
Eva. T1	370	367	138,688	542,086	16
Eva. T2	370 (194)	367 (173)	100,708	395,924	16

with diverse devices. This is in contrast to the source database (VCTK [12]) of previous challenges, which contains data collected from 100 speakers in a hemi-anechoic chamber.

The second major evolution is the use of stronger spoofing attacks. In addition to using the latest TTS and VC algorithms, spoofing attacks are tuned to fool not only ASV, but also CM surrogate sub-systems. This is a key difference to previous ASVspoof databases which were constructed by verifying that spoofing attack data were successful in manipulating an ASV sub-system only. Adversarial attacks, applied to augment the threat of spoofing attacks, were created using Malafide [13] and Malacopula [14] filtering. The former aims to compromise the performance of CMs, while the latter escalates the threat of spoofed data to ASV. Last, codecs, including neural-network-based variants, are applied to both bona fide and spoofed data.

The database was constructed in three steps with the help of two groups of data contributors. First, the MLS English dataset was divided into three disjoint partitions: A, B, and C. Data contributors in the first group used partition A to build TTS systems. We used the resulting data to train surrogate ASV and CM systems (described in Section 4). Data contributors of the second group then used partition B to build new TTS and VC systems. The latter are tuned with the use of surrogate CM and ASV systems to produce cloned voices which successfully fool both sub-systems. Finally, tuned TTS and VC systems were used to clone the voices of target speakers in partition C. A subset of TTS and VC systems were further combined with Malafide and Malacopula filtering. Note that, to avoid potential data leakage, spoofing attacks and surrogate systems were built with privileged protocols which were not shared with challenge participants.

Bona fide data in the training set is sourced from speakers in partition A, whereas spoofed data is generated using TTS systems built by the first group of data contributors. Bona fide data in the development and evaluation sets are sourced from partition C, whereas spoofed data is created by the second group of data contributors. The speakers in the ASVspoof 5 challenge training, development, and evaluation sets are disjoint. A summary of key statistics is shown in Table 1.[1]

The spoofing attacks in training, development, and evaluation sets are also disjoint. A brief summary of each attack is shown in Table 2. In addition to legacy TTS and VC algorithms (e.g., MaryTTS [16]), ASVspoof 5 spoofing attacks were generated using the latest DNN-based methods (e.g., ZMM-TTS [17]). Two pre-trained systems, namely YourTTS [18] and XTTS [19], were also used for the cloning of target speaker voices in a zero-shot manner.

[1]The MLS English dataset is derived from the same source as Librispeech [15]. Because challenge participants in the open condition were permitted to use models pre-trained using Librispeech, we removed all data collected from speakers appearing in the evaluation set when they also appear in Librispeech.

Table 2: Summary of spoofing attacks. A01-A08, A09-A16, and A17-A32 are in training, development, and evaluation sets, respectively. AT denotes adversarial attack using Malafide, Malacopula, or both.

ID	Type	Algorithm	ID	Type	Algorithm
A01	TTS	GlowTTS [20]	A17	TTS	ZMM-TTS [17]
A02	TTS	variant of A01	A18	AT	A17+Malafide
A03	TTS	variant of A01	A19	TTS	MaryTTS [16]
A04	TTS	GradTTS [21]	A20	AT	A12+Malafide
A05	TTS	variant of A04	A21	TTS	A09+BigVGAN [22]
A06	TTS	variant of A04	A22	TTS	variant of A09 [23]
A07	TTS	FastPitch [24]	A23	AT	A09+Malafide
A08	TTS	VITS [25]	A24	VC	In-house ASR-based
A09	TTS	ToucanTTS [26]	A25	VC	DiffVC [27]
A10	TTS	A09+HifiGANv2 [28]	A26	VC	A16+original genuine noise
A11	TTS	Tacotron2 [29]	A27	AT	A26+Malacopula
A12	TTS	In-house unit-select	A28	TTS	Pre-trained YourTTS [18]
A13	VC	StarGANv2-VC [30]	A29	TTS	Pre-trained XTTS [19]
A14	TTS	YourTTS [18]	A30	AT	A18+Malafide+Malacopula
A15	VC	VAE-GAN [31]	A31	AT	A22+Malacopula
A16	VC	In-house ASR-based	A32	AT	A25+Malacopula

Table 3: Summary of codec and compression conditions in evaluation sets of Track 1 (☆) and Track 2 (★).

	Codec	Bandwidth	Bitrate range	Usage
C00	-	16 kHz	-	☆ ★
C01	opus	16 kHz	6.0 - 30.0	☆ ★
C02	amr	16 kHz	6.6 - 23.05	☆ ★
C03	speex	16 kHz	5.75 - 34.20	☆ ★
C04	Encodec [32]	16 kHz	1.5 - 24.0	☆
C05	mp3	16 kHz	45 - 256	☆
C06	m4a	16 kHz	16 - 128	☆
C07	mp3+Encodec	16 kHz	varied	☆
C08	opus	8 kHz	4.0 - 20.0	☆ ★
C09	amr	8 kHz	4.75 - 12.20	☆ ★
C10	speex	8 kHz	3.95 - 24.60	☆ ★
C11	varied	8 kHz	varied	☆ ★

To evaluate the performance of CM and SASV systems when both bona fide and spoofed data are (lossy) encoded or compressed, evaluation data was treated with the set of codecs listed in Table 3. For condition C00, there is no encoding or compression. For all other conditions, bona fide and spoofed utterances were treated with one of the codecs C01-C11. C01-C07 operate with a 16 kHz sampling rate, while C08-C11 operate in an 8 kHz narrow band setting. To create narrow band data, bona fide and spoofed data were down-sampled to 8 kHz, processed with the codec, and then up-sampled to 16 kHz. All data are saved in FLAC format with a sampling rate of 16 kHz. For all utterances in the evaluation data, leading and trailing non-speech segments in the evaluation set utterances were removed.

Participants in the closed condition of both tracks were required to build their systems using data in the training and development sets only. For both tracks, participants in the open condition were permitted to use external training data, but only under the condition that there is no overlap with the challenge database. The use of pre-trained speech foundation models built using a selection of publicly available databases [33, §4.2] was allowed. The evaluation sets for both tracks comprise the same set of utterances, except that for the four codec conditions highlighted in Table 3 which were excluded in Track 2.

3. Performance measures

This section provides a brief summary of the performance measures used in the two challenge tracks.

3.1. Track 1: from EER to DCF

Systems submitted to Track 1 were requried to assign a real-valued bona fide-spoof detection score to each utterance. Different to past ASVspoof editions for which the EER was used as the primary metric, ASVspoof 5 adopts the *normalized detection cost function* (DCF) [8] for Track 1. While further details are available in [33, Appendix], the DCF has the simple form

$$\text{DCF}(\tau_{\text{cm}}) = \beta \cdot P_{\text{miss}}^{\text{cm}}(\tau_{\text{cm}}) + P_{\text{fa}}^{\text{cm}}(\tau_{\text{cm}}), \tag{1}$$

where $P_{\text{miss}}^{\text{cm}}(\tau_{\text{cm}})$ is the miss rate (false rejection rate for bona fide utterances) and $P_{\text{fa}}^{\text{cm}}(\tau_{\text{cm}})$ is the false alarm rate (false acceptance rate for spoofed utterances). Both are functions of a detection threshold, τ_{cm}, and the constant β in (1) is defined as

$$\beta = \frac{C_{\text{miss}}}{C_{\text{fa}}} \cdot \frac{1 - \pi_{\text{spf}}}{\pi_{\text{spf}}}, \tag{2}$$

where C_{miss} and C_{fa} are, respectively, the costs of misses and false alarms, and where π_{spf} is the asserted prior probability of spoofing attacks.[2] For the scenario envisioned in Track 1 we assume that, compared to spoofed utterances, bona fide speech utterances are, in general, far more likely in practice (low π_{spf}). But, when encountered but not detected, the relative cost is high. We set $C_{\text{miss}} = 1$, $C_{\text{fa}} = 10$, $\pi_{\text{spf}} = 0.05$, which gives $\beta \approx 1.90$.

The normalized DCF in (1) is used to compute both *minimum* and *actual* DCFs. The former is the primary metric for Track 1, defined as $\text{minDCF} = \min_{\tau_{\text{cm}}} \text{DCF}(\tau_{\text{cm}})$. The latter, $\text{actDCF} = \text{DCF}(\tau_{\text{Bayes}})$, is the DCF evaluated at a fixed threshold $\tau_{\text{Bayes}} = -\log(\beta)$ under the assumption that detection scores can be interpreted as log-likelihood ratios (LLRs). Whereas the minDCF measures performance using an 'oracle' threshold (set based on ground-truth), the actDCF measures the realised cost obtained by setting the threshold to τ_{Bayes} [8]. Note that this is meaningful only when scores can be interpreted as calibrated LLRs [34, 35]. Similar to past challenge editions, the submission of LLR scores was not *required*—rather, it was *encouraged* for the first time.[3]

Another complementary metric, the *cost of log-likelihood ratios* (C_{llr}) [34], was used to assess the quality of detection scores when interpreted as LLRs

$$C_{\text{llr}} = \frac{1}{2\log 2}\left(\frac{1}{|\mathscr{B}|}\sum_{s_i \in \mathscr{B}} \log\left(1+e^{-s_i}\right) + \frac{1}{|\mathscr{S}|}\sum_{s_j \in \mathscr{S}} \log\left(1+e^{s_j}\right)\right), \tag{3}$$

where $\mathscr{B} = \{s_i\}$ and $\mathscr{S} = \{s_j\}$ denote, respectively, the sets of bona fide and spoofed trial scores. The lower C_{llr}, the better calibrated (and more discriminative) the scores. In addition to minDCF, actDCF, and C_{llr}, the EER is also reported.

3.2. Track 2: from SASV-EER to a-DCF

For Track 2, participants could submit either single real-valued SASV scores or a triplet of scores which, in addition to SASV scores, contains spoofing (CM sub-system) and speaker (ASV sub-system) detection scores. While the former can be produced by any model architecture which outputs a single detection score, the latter assumes a specific tandem (cascade) architecture [10] consisting of two clearly-identified CM and ASV sub-systems. In this case, SASV scores are generated by combining sub-systems outputs (e.g., embeddings or scores) using an arbitrary combination strategy designed by the participants.

For both types of submission, SASV scores are used to compute the primary challenge metric. Track 2 takes a step forward from EER-based metrics used in the first SASV challenge [6] to DCF-based metrics. Building upon the two-class DCF (1), a *normalized architecture-agnostic* detection cost function (a-DCF) [9] was recently proposed and is defined as

$$\begin{aligned}\text{a-DCF}(\tau_{\text{sasv}}) =& \alpha P_{\text{miss}}^{\text{sasv}}(\tau_{\text{sasv}}) + (1-\gamma) P_{\text{fa,non}}^{\text{sasv}}(\tau_{\text{sasv}}) \\ &+ \gamma P_{\text{fa,spf}}^{\text{sasv}}(\tau_{\text{sasv}}),\end{aligned} \tag{4}$$

where $P_{\text{miss}}^{\text{sasv}}(\tau_{\text{sasv}})$ is the ASV miss (target speaker false rejection) rate and where $P_{\text{fa,non}}^{\text{sasv}}(\tau_{\text{sasv}})$ and $P_{\text{fa,spf}}^{\text{sasv}}(\tau_{\text{sasv}})$ are the false alarm (false acceptance) rates for non-targets and spoofing attacks, respectively. All three error rates are functions of an SASV threshold τ_{sasv}. The constants α and γ are given by

$$\begin{aligned}\alpha &= \frac{C_{\text{miss}}\pi_{\text{tar}}}{C_{\text{fa,non}}\pi_{\text{non}} + C_{\text{fa,spf}}\pi_{\text{spf}}}, \\ \gamma &= \frac{C_{\text{fa,spf}}\pi_{\text{spf}}}{C_{\text{fa,non}}\pi_{\text{non}} + C_{\text{fa,spf}}\pi_{\text{spf}}},\end{aligned} \tag{5}$$

where C_{miss}, $C_{\text{fa,non}}$, and $C_{\text{fa,spoof}}$ are the costs of a miss, the false acceptance of a non-target speaker, and the false acceptance of a spoofing attack, and where π_{tar}, π_{non}, and π_{spoof} are the asserted priors of targets, non-targets (zero-effort impostors), and spoofing attacks. The assumptions are similar to those for Track 1. We set $\pi_{\text{tar}} = 0.9405$, $\pi_{\text{non}} = 0.0095$, $\pi_{\text{spf}} = 0.05$, $C_{\text{miss}} = 1$ and $C_{\text{fa,non}} = C_{\text{fa,spf}} = 10$. This gives $\alpha \approx 1.58$ and $\gamma \approx 0.84$. The primary metric for Track 2 is the minimum a-DCF, $\text{min a-DCF} = \min_{\tau_{\text{sasv}}} \text{a-DCF}(\tau_{\text{sasv}})$.

For submissions which provide clearly-identified ASV and CM sub-systems, the *ASV-constrained minimum tandem detection cost function* (t-DCF) [10] and the *tandem equal error rate* (t-EER) [11] are also reported. Whereas the former has served as the primary metric since ASVspoof 2019, the latter provides a complementary parameter-free measure of class discrimination. The t-DCF metric is computed using the same costs and priors as above and using ASV scores produced by a common ASV system (see Section 4) in place of scores provided the participant. This allows computation of the minimum 'ASV-constrained' t-DCF in the same way as for the previous ASVspoof challenges and enables the comparison of different CM sub-systems when they are combined with a common ASV sub-system.

For computation of the t-EER metric, both the CM and ASV sub-system scores are used to obtain a single *concurrent t-EER* value, denoted by $\text{t-EER}_{\times}$. It has a simple interpretation as the error rate at a unique *pair* of ASV and CM thresholds, $\boldsymbol{\tau}^{\times} := (\tau_{\text{asv}}^{\times}, \tau_{\text{cm}}^{\times})$, at which the miss rate and the two types of false alarm rates (one for spoofing attacks, the other for non-targets) are equal: $\text{t-EER}_{\times} = P_{\text{miss}}^{\text{tdm}}(\boldsymbol{\tau}^{\times}) = P_{\text{fa,non}}^{\text{tdm}}(\boldsymbol{\tau}^{\times}) = P_{\text{fa,spoof}}^{\text{tdm}}(\boldsymbol{\tau}^{\times})$. The superscript 'tdm' is used to emphasize the assumed tandem architecture. The t-EER can be seen as a generalisation of the conventional two-class, single system EER which provides an application-agnostic discrimination measure.

[2] Since we have only two classes, it follows that $1 - \pi_{\text{spf}}$ is the asserted prior of the bona fide class.

[3] Raw detection scores can be post-processed into LLRs using implementations such as [35] in order to reduce actDCF. Note, however, that any order-preserving score calibration does not affect the primary minDCF metric.

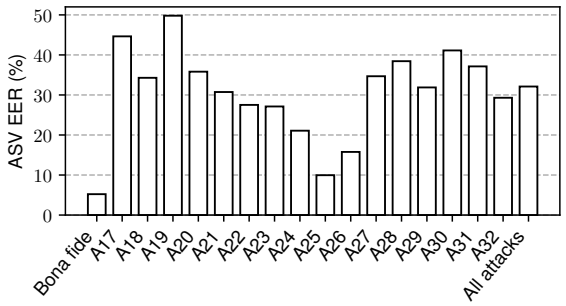


Figure 1: ASV EERs for the common ASV system and evaluation data. Results are pooled over the set of codec conditions.

4. Common ASV, baseline and surrogate systems

4.1. Common ASV system

The common ASV system uses an ECAPA-TDNN speaker encoder [36] and cosine similarity scoring. The ECAPA-TDNN model is trained using the training partitions of the VoxCeleb 1 and 2 databases [37]. Cosine scores are subsequently normalised using s-norm. Figure 1 illustrates ASV EERs for the evaluation data. When discriminating between bona fide target and non-target data (leftmost bar), the EER is 5% . However, the EERs are much higher when bona fide non-target data is replaced with spoofing attacks. Note that, although A25 is the least effective attack, it proves more threatening when enhanced in the form of an adversarial attack A32.

4.2. Baseline systems

For Track 1 there are two CM baseline systems: RawNet2 [38] (B01) and AASIST [39] (B02). Both systems are end-to-end,operating directly on raw waveforms of 4 seconds duration (64,000 samples). RawNet2 is composed of a fixed bank of 20 sinc filters and six residual blocks followed by gated recurrent units, which convert frame-level representations to utterance-level representations. Output scores are generated using fully-connected layers. AASIST uses a RawNet2-based encoder [38] to extract spectro-temporal features from the input waveform. A spectro-temporal heterogeneous graph attention layers and max graph operations are then used to integrate temporal and spectral representations. CM output scores are generated using a readout operation and a linear output layer. Both baselines were trained with a weighted cross-entropy loss for binary classification.

There are two baselines for Track 2: a fusion-based system [6] (B03) and a single integrated system [40] (B04). B03, adopted from the SASV 2022 challenge baseline, is a fusion of the common ASV system and the Track 1 AASIST baseline, using an LLR-based fusion tool [41]. B04, which is based on MFA-Conformer [42], extracts a single embedding from the input waveform and produces a single SASV score. It is trained in three stages: speaker classification-based pre-training, copy synthesis training [43] with adapted SASV loss functions, and in-domain fine-tuning. VoxCeleb and copy synthesis data are used in the first and second stages, respectively. In-domain fine-tuning is conducted using ASVspoof 5 training data.

Source codes for all baselines are accessible from the ASVspoof 5 Github repository.[4]

[4] github.com/asvspoof-challenge/asvspoof5

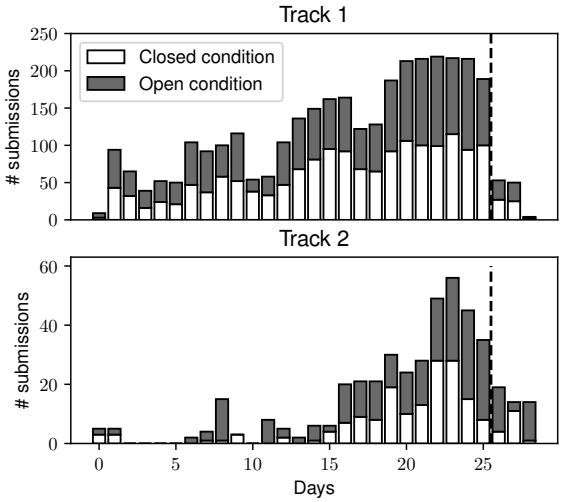


Figure 2: A *stacked* bar chart showing the number of CodaLab submissions to Tracks 1 and 2 for the 26-day *progress* and 3-day *evaluation* phases.

4.3. Surrogate systems

The surrogate ASV system is based on ECAPA-TDNN and a probabilistic linear discriminant analysis scoring backend [44]. Surrogate CM systems include AASIST, RawNet2, and LCNNs with LFCC features, all of which are trained using bona fide data in MLS partition A and spoofing attacks created by the first group of data contributors (see Section 2), i.e. without attacks contained in either the development or evaluation sets.

5. Evaluation platform

ASVspoof 5 used the CodaLab website through which participants could submit detection scores and receive results. The challenge was run in two phases (with an additional post-evaluation phase not addressed in this paper). During the first 26-day *progress* phase, participants could make up to four submissions per day. Results derived from an evaluation subset (the progress subset) were made available to participants who could then opt to submit their results to an anonymised leaderboard. The evaluation phase ran for only a few days, during which participants could make only a single submission. Evaluation submissions were evaluated using the whole evaluation set.

Figure 2 illustrates the number of submissions during the progress and evaluation phases. For Track 1, there were a comparable number of submissions to closed and open conditions. In contrast, for Track 2, the number of submissions to the open condition outstripped those to the closed condition, possibly an indication of the need for additional data to support the training of SASV systems.

6. Challenge results

6.1. Track 1

Results for Track 1 are listed in Table 4. The baseline systems achieve minDCFs no lower than 0.7 and EERs no lower than 29%. Even if RawNet2 and AASIST both achieve promising results in the case of previous ASVspoof challenge databases, results for the ASVspoof 5 database are far worse, and are likely caused by the use of non-studio-quality source data as well as

Table 4: Track 1 evaluation results. Ensemble systems and single systems are marked by • and ∘, respectively. Open-condition submissions using and not using pre-trained self-supervised models are marked by ▲ and △, respectively. The absence of a Team ID indicates submissions for which a system description was not received. Submissions made after the initial deadline are underscored.

Closed condition

	#	ID	minDCF	actDCF	C_{llr}	EER	#	ID	minDCF	actDCF	C_{llr}	EER
•	1	T32	0.2436	0.9956	0.9458	8.61	18	-	0.5990	0.9666	6.6313	24.12
•	2	T47	0.2660	0.3380	0.6091	9.18	19	-	0.6086	0.6091	0.8265	28.65
•	3	T24	0.2975	0.2976	0.4182	10.43	• 20	T07	0.6285	1.0000	1.0752	25.47
•	4	T45	0.3948	1.0000	0.8515	14.33	• 21	T27	0.6339	1.0937	1.0808	26.17
•	5	T13	0.4025	0.4218	0.5238	14.75	22	-	0.6463	0.8388	2.3251	26.45
	6	-	0.4079	0.4299	0.5512	14.16	• 23	T41	0.6543	0.7641	0.9184	26.28
	7	-	0.4390	0.6332	0.8531	17.09	∘ 24	T06	0.6598	1.0000	1.1159	28.41
•	8	T46	0.4783	1.0000	1.0509	20.45	25	-	0.6617	0.9894	0.9562	27.31
•	9	T23	0.5312	1.0000	1.1171	20.13	∘ 26	T14	0.6618	0.9307	2.4858	25.32
	10	-	0.5340	1.0000	1.0228	19.10	27	-	0.6989	0.7006	1.6935	31.15
	11	-	0.5357	0.9533	3.3069	22.67	∘ 28	**B02**	0.7106	0.9298	4.0014	29.12
•	12	T35	0.5505	1.0000	1.1435	23.42	∘ 29	T44	0.7997	1.0000	1.2774	35.15
	13	-	0.5809	0.8537	4.0994	23.34	30	-	0.8165	1.0000	1.1236	44.94
∘	14	T48	0.5813	0.9354	3.1923	23.63	∘ 31	**B01**	0.8266	0.9922	4.0935	36.04
∘	15	T19	0.5891	0.6883	1.3277	24.59	• 32	T54	0.8624	1.0000	1.1221	39.68
	16	-	0.5895	1.0000	0.9351	23.93	∘ 33	T53	0.9744	1.0539	2.4977	44.94
	17	-	0.5899	0.7470	1.3798	22.58						

Open condition

	#	ID	minDCF	actDCF	C_{llr}	EER	#	ID	minDCF	actDCF	C_{llr}	EER
•▲	1	T45	0.0750	1.0000	0.7923	2.59	18	-	0.1949	0.2438	0.7028	7.05
•▲	2	T36	0.0936	1.0000	0.8874	3.41	19	-	0.1966	1.0000	0.9327	6.80
•▲	3	T27	0.0937	0.1375	0.1927	3.42	•▲ 20	T33	0.2021	0.6028	0.5560	7.01
•▲	4	T23	0.1124	1.0000	0.9179	4.16	21	-	0.2148	1.0000	0.8124	7.43
•▲	5	T43	0.1149	0.5729	0.9562	4.04	•▲ 22	T51	0.2236	1.0000	0.8011	7.72
•▲	6	T13	0.1301	0.1415	0.3791	4.50	•▲ 23	T46	0.2245	1.0000	1.0308	9.36
•▲	7	T06	0.1348	0.2170	0.3096	5.02	24	-	0.2573	1.0000	0.9955	9.28
	8	-	0.1414	0.5288	0.6149	4.89	25	-	0.2642	0.7037	2.1892	10.32
∘▲	9	T31	0.1499	0.2244	0.5559	5.56	•△ 26	T47	0.2660	0.3321	0.4932	9.18
•▲	10	T29	0.1549	0.2052	0.7288	5.37	27	-	0.2668	0.2923	0.6194	9.59
•▲	11	T35	0.1611	1.0000	1.0384	5.93	•▲ 28	T41	0.3010	0.3095	0.4773	10.45
	12	-	0.1665	0.1669	0.2351	5.77	29	-	0.4121	0.4266	0.7185	14.25
•▲	13	T21	0.1728	0.2392	0.9498	6.01	•▲ 30	T02	0.4845	1.0000	0.9332	17.08
∘▲	14	T17	0.1729	1.0000	2.3217	5.99	∘△ 31	T15	0.5112	0.6723	0.8858	22.24
∘▲	15	T19	0.1743	0.3087	0.4757	6.06	32	-	0.6584	0.7451	1.1404	22.90
	16	-	0.1840	1.0000	0.8764	6.35	33	-	0.7969	1.0000	0.9920	35.72
	17	-	0.1933	1.0000	0.8342	6.67	∘△ 34	T53	0.9744	1.0539	2.4977	44.94

more advanced spoofing and adversarial attacks.

Encouragingly, most submissions to the closed condition outperform the baselines in terms of minDCF. The top-5 submissions obtain minDCFs below 0.5 and EERs below 15%, a relative improvement over the baselines of ~50%. Similar to the trend observed in previous challenge editions, submissions using an ensemble of sub-systems tend to perform better.

Unsurprisingly, minDCF and EER values for the open condition are lower than those for the closed condition. Notably, most of the top-performing submissions use features extracted using pre-trained, self-supervised learning (SSL) models, e.g., wav2vec 2.0 (base version) [45].

Despite the encouraging results, the top systems for both conditions obtain actDCF values close or equal to 1.0. This is because system outputs are 'normalized' to between 0 and 1 rather than being calibrated to approximate LLRs. Scores are above the optimal Bayes decision threshold specified by the priors and decision costs, which leads to $P_{\text{miss}}^{\text{cm}}(\tau_{\text{cm}}) = 0, P_{\text{fa}}^{\text{cm}}(\tau_{\text{cm}}) = 1$, and actDCF equal to 1.0. C_{llr} values are also high, again a sign of poor calibration. In contrast, some systems, such as T24 under the closed condition, are better calibrated.

6.2. Track 2

Results for Track 2 are listed in Table 5. The design of SASV solutions is perhaps more technically demanding than that of stand-alone CMs. This might account for the lower number of submissions to Track 2. Performance for B03 is not dissimilar to that of the reference system (REF) which is the same as B03 except for the use of a CM sub-system which produces random outputs. guessing CM sub-system. This indicates that the CM sub-system of B03 does not provide information which

Table 5: Track 2 evaluation results. Submissions with only SASV scores are not evaluated using min t-DCF and t-EER. Submissions using a system ensemble and a single system are marked by • and ∘, respectively. Open-condition submissions using and not using pre-trained self-supervised models are marked by ▲ and △, respectively. The absence of a Team ID indicates submissions for which a system description was not received. Submissions made after the deadline are underscored. REF denotes the organisers' ASV (§ 4) without a CM.

Closed condition

	#	ID	min a-DCF	min t-DCF	t-EER	#	ID	min a-DCF	min t-DCF	t-EER
•	1	T45	0.2814	-	-	• 9	T23	0.4513	0.8279	49.34
•	2	T24	0.2954	0.6175	9.58	10	-	0.5130	-	-
•	3	T47	0.3173	0.5261	7.49	∘ 11	**B04**	0.5741	-	-
	4	-	0.3542	-	-	12	-	0.6209	0.9073	25.39
	5	-	0.3744	-	-	∘ 13	**B03**	0.6806	0.9295	28.78
	6	-	0.3893	0.7783	20.85	∘ 14	**REF**	0.6869	-	-
	7	-	0.3896	-	-	15	-	0.8985	-	-
	8	-	0.3971	0.7007	15.09					

Open condition

	#	ID	min a-DCF	min t-DCF	t-EER	#	ID	min a-DCF	min t-DCF	t-EER
•▲	1	T45	0.0756	-	-	7	-	0.1797	0.5430	8.39
•▲	2	T39	0.1156	0.4584	4.32	8	-	0.3896	-	-
•▲	3	T36	0.1203	0.4291	4.54	9	-	0.4581	-	-
•▲	4	T06	0.1295	0.4372	5.43	∘△ 10	**REF**	0.6869	-	-
∘▲	5	T29	0.1410	0.4690	5.48	11	-	0.9134	-	-
•▲	6	T23	0.1492	0.4075	4.63					

is useful to the rejection of spoofing attacks. The single integrated B04 baseline performs better. However, these results do not show that fusion-based solutions are inferior; all of the top-performing submissions are based upon the fusion of ASV and CM sub-systems, including T45.

Most submissions outperform the baselines. For the top-3 submissions to the closed condition, the improvements are ∼50% relative to the best baseline in terms of min a-DCF. Similar to findings for Track 1, submissions to the open condition achieve better performance and the use of SSL-based features is common among the top-performing submissions.

7. Conclusions

We present an outline of the ASVspoof 5 challenge which is designed to support the evaluation of both stand-alone speech spoofing and deepfake detection and SASV solutions. The fifth edition was considerably more complex than its predecessors, and included not only a new task, but also more challenging crowdsourced data collected under variable conditions, spoofing attacks generated with a variety of contemporary algorithms tuned to fool surrogate ASV and CM sub-systems, and new adversarial attacks. Despite the use of lower-quality data to create spoofs and deepfakes, detection performance for the baseline systems, all top-performing systems reported in recent years, is relatively poor. Encouragingly, results for most challenge submissions outperform the challenge baselines, sometimes by a substantial margin. Results also reveal the hitherto ignored issue of score calibration, an essential consideration if detection solutions are deployed in practical scenarios. With a particularly tight schedule for ASVspoof 5, more detailed analyses will be presented at the workshop and reported in future work.

8. Acknowledgements

The ASVspoof 5 organising committee expresses its gratitude and appreciation to the challenge participants. For reasons of anonymity, they could not be identified in this article. Subject to the publication of their results and prior approval, they will be cited or otherwise acknowledged in future work.

The ASVspoof 5 organising committee extends its sincere gratitude to data contributors (in alphabetic order): Cheng Gong, Tianjin University; Chengzhe Sun, Shuwei Hou, Siwei Lyu, University at Buffalo, State University of New York; Florian Lux, University of Stuttgart; Ge Zhu, Neil Zhang, Yongyi Zang, University of Rochester; Guo Hanjie and Liping Chen, University of Science and Technology of China; Hengcheng Kuo and Hung-yi Lee, National Taiwan University; Myeonghun Jeong, Seoul National University; Nicolas Muller, Fraunhofer AISEC; Sébastien Le Maguer, University of Helsinki; Soumi Maiti, Carnegie Mellon University; Yihan Wu, Renmin University of China; Yu Tsao, Academia Sinica; Vishwanath Pratap Singh, University of Eastern Finland; Wangyou Zhang, Shanghai Jiaotong University.

The committee would like to acknowledge A*STAR (Singapore) for sponsoring its use of the CodaLab platform, in addition to Pindrop (USA) and KLASS Engineering and Solutions (Singapore) for sponsoring the ASVspoof 2024 Workshop. This work is also partially supported by JST, PRESTO Grant Number JPMJPR23P9, Japan and with funding received from the French Agence Nationale de la Recherche (ANR) via the BRUEL (ANR-22-CE39-0009) and COMPROMIS (ANR-22-PECY-0011) projects. This work was also partially supported by the Academy of Finland (Decision No. 349605, project SPEECHFAKES). Part of this work used the TSUBAME4.0 supercomputer at Tokyo Institute of Technology.

9. References

[1] Zhizheng Wu, Tomi Kinnunen, Nicholas Evans, Junichi Yamagishi, Cemal Hanilçi, Md. Sahidullah, and Aleksandr Sizov, "ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Proc. Interspeech*, 2015, pp. 2037–2041.

[2] Tomi Kinnunen, Md. Sahidullah, Hector Delgado, Massimiliano Todisco, Nicholas Evans, Junichi Yamagishi, and Kong-Aik Lee, "The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Interspeech*, 2017, pp. 2–6.

[3] Massimiliano Todisco, Xin Wang, Ville Vestman, Md. Sahidullah, Hector 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.

[4] Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, and Hector Delgado, "ASVspoof 2021: Accelerating progress in spoofed and deepfake speech detection," in *Proc. ASVspoof Challenge Workshop*, 2021, pp. 47–54.

[5] Xuechen Liu, Xin Wang, Md Sahidullah, Jose Patino, Hector Delgado, Tomi Kinnunen, Massimiliano Todisco, Junichi Yamagishi, Nicholas Evans, Andreas Nautsch, and Kong Aik Lee, "ASVspoof 2021: Towards Spoofed and Deepfake Speech Detection in the Wild," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2507–2522, 2023.

[6] Jee-weon Jung, Hemlata Tak, Hye-jin Shim, Hee-Soo Heo, Bong-Jin Lee, Soo-Whan Chung, Ha-Jin Yu, Nicholas Evans, and Tomi Kinnunen, "SASV 2022: The First Spoofing-Aware Speaker Verification Challenge," in *Proc. Interspeech*, 2022, pp. 2893–2897.

[7] Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, and Ronan Collobert, "MLS: A Large-Scale Multilingual Dataset for Speech Research," in *Proc. Interspeech*, 2020, pp. 2757–2761.

[8] NIST, *NIST 2020 CTS Speaker Recognition ChallengeEvaluation Plan*, 2020.

[9] Hye-jin Shim, Jee-weon Jung, Tomi Kinnunen, et al., "a-DCF: an architecture agnostic metric with application to spoofing-robust speaker verification," in *Proc. Speaker Odyssey*, 2024, pp. 158–164.

[10] Tomi Kinnunen, Héctor Delgado, Nicholas Evans, 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.

[11] Tomi H. Kinnunen, Kong Aik Lee, Hemlata Tak, et al., "t-EER: Parameter-free tandem evaluation of countermeasures and biometric comparators," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 2622–2637, 2024.

[12] Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald, "CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (version 0.92)," 2019.

[13] Michele Panariello, Wanying Ge, Hemlata Tak, Massimiliano Todisco, and Nicholas Evans, "Malafide: a novel adversarial convolutive noise attack against deepfake and spoofing detection systems," in *Proc. Interspeech*, 2023, pp. 2868–2872.

[14] Massimiliano Todisco, Michele Panariello, Xin Wang, Hector Delgado, Kong-Aik Lee, and Nicholas Evans, "Malacopula: Adversarial automatic speaker verification attacks using a neural-based generalised hammerstein model," in *Proc. ASVspoof Workshop 2024*, 2024.

[15] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[16] Ingmar Steiner and Sébastien Le Maguer, "Creating new language and voice components for the updated marytts text-to-speech synthesis platform," in *Proc. LREC*, 2018, pp. 3171–3175.

[17] Cheng Gong, Xin Wang, Erica Cooper, Dan Wells, Longbiao Wang, Jianwu Dang, Korin Richmond, and Junichi Yamagishi, "ZMM-TTS: Zero-shot multilingual and multispeaker speech synthesis conditioned on self-supervised discrete speech representations," *arXiv preprint arXiv:2312.14398*, 2023.

[18] Edresson Casanova, Julian Weber, Christopher D Shulby, Arnaldo Candido Junior, Eren Gölge, and Moacir A Ponti, "Yourtts: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone," in *Proc. ICML*, 2022, pp. 2709–2720.

[19] Edresson Casanova, Kelly Davis, Eren Gölge, Görkem Göknar, Iulian Gulea, Logan Hart, Aya Aljafari, Joshua Meyer, Reuben Morais, Samuel Olayemi, et al., "XTTS: A massively multilingual zero-shot text-to-speech model," *Proc. Interspeech*, 2024.

[20] Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon, "Glow-TTS: A generative flow for text-to-speech via monotonic alignment search," in *Proc. NeurIPS*, 2020, vol. 33, pp. 8067–8077.

[21] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov, "Grad-TTS: A diffusion probabilistic model for text-to-speech," in *Proc. ICML*, 2021, pp. 8599–8608.

[22] Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon, "BigVGAN: A universal neural vocoder with large-scale training," in *Proc. ICLR*, 2022.

[23] Florian Lux, Julia Koch, and Ngoc Thang Vu, "Exact Prosody Cloning in Zero-Shot Multispeaker Text-to-Speech," in *Proc. SLT*, 2023, pp. 962–969.

[24] Adrian Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *Proc. ICASSP*, 2021, pp. 6588–6592.

[25] Jaehyeon Kim, Jungil Kong, and Juhee Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. ICML*, 2021, pp. 5530–5540.

[26] Florian Lux, Julia Koch, and Ngoc Thang Vu, "Low-resource multilingual and zero-shot multispeaker TTS," in *Proc. AACL*, 2022, pp. 741–751.

[27] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, Mikhail Kudinov, and Jiansheng Wei, "Diffusion-based voice conversion with fast maximum likelihood sampling scheme," in *Proc. ICLR*, 2022.

[28] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Proc. NeurIPS*, 2020, vol. 33, pp. 17022–17033.

[29] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al., "Natural TTS synthesis by conditioning wavenet on Mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4779–4783.

[30] Yinghao Aaron Li, Ali Zare, and Nima Mesgarani, "StarGANv2-VC: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion," in *Proc. Interspeech*, 2021, pp. 1349–1353.

[31] Ehab A. AlBadawy and Siwei Lyu, "Voice Conversion Using Speech-to-Speech Neuro-Style Transfer," in *Proc. Interspeech*, 2020, pp. 4726–4730.

[32] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi, "High fidelity neural audio compression," *Transactions on Machine Learning Research*, 2023, Featured Certification, Reproducibility Certification.

[33] Hector Delgado, Nicholas Evans, Jee-weon Jung, Tomi Kinnunen, Ivan Kukanov, Kong-Aik Lee, Xuechen Liu, Hye-jin Shim, Md Sahidullah, Hemlata Tak, Massimiliano Todisco, Xin Wang, and Junichi Yamagishi, "ASVspoof 5 evaluation plan (phase 2)," `https://www.asvspoof.org/file/ASVspoof5___Evaluation_Plan_Phase2.pdf`, v0.6, accessed 23-July-2024.

[34] Niko Brümmer and Johan du Preez, "Application-independent evaluation of speaker detection," *Computer Speech & Language*, vol. 20, no. 2, pp. 230–275, 2006.

[35] Luciana Ferrer, "Calibration tutorial," https://github.com/luferrer/CalibrationTutorial, 2024.

[36] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.

[37] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "VoxCeleb: A Large-Scale Speaker Identification Dataset," in *Proc. Interspeech*, 2017, pp. 2616–2620.

[38] Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher, "End-to-end anti-spoofing with RawNet2," in *Proc. ICASSP*. IEEE, 2021, pp. 6369–6373.

[39] Jee-weon Jung, Hee-Soo Heo, Hemlata Tak, Hye-jin Shim, Joon Son Chung, Bong-Jin Lee, Ha-Jin Yu, and Nicholas Evans, "AASIST: Audio anti-spoofing using integrated spectro-temporal graph attention networks," in *Proc. ICASSP*, 2022, pp. 6367–6371.

[40] Sung Hwan Mun, Hye-jin Shim, Hemlata Tak, Xin Wang, Xuechen Liu, Md Sahidullah, Myeonghun Jeong, Min Hyun Han, Massimiliano Todisco, Kong Aik Lee, et al., "Towards single integrated spoofing-aware speaker verification embeddings," in *Proc. Interspeech*, 2023, pp. 3989–3993.

[41] Xin Wang, Tomi Kinnunen, Lee Kong Aik, Paul-Gauthier Noe, and Junichi Yamagishi, "Revisiting and improving scoring fusion for spoofing-aware speaker verification using compositional data analysis," in *Proc. Interspeech*, 2024.

[42] Yang Zhang, Zhiqiang Lv, Haibin Wu, Shanshan Zhang, Pengfei Hu, Zhiyong Wu, Hung-yi Lee, and Helen Meng, "MFA-conformer: Multi-scale feature aggregation conformer for automatic speaker verification," in *Proc. Interspeech*, 2022, pp. 306–310.

[43] Xin Wang and Junichi Yamagishi, "Spoofed training data for speech spoofing countermeasure can be efficiently created using neural vocoders," in *Proc. ICASSP*, 2023.

[44] Simon JD Prince and James H Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. ICCV*, 2007, pp. 1–8.

[45] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NuerIPS*, 2020, vol. 33, pp. 12449–12460.