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

Data augmentations for audio deepfake detection for the ASVspoof5 closed condition

Raphaël Duroselle[1], Olivier Boeffard[1], Adrien Courtois[1], Hubert Nourtel[2], Pierre Champion[1], Heiko Agnoli[1], Jean-François Bonastre[1]

[1]Inria Défense et Sécurité / LR2, [2]Storyzy
France

firstname.name@inria.fr, firstname.name@storyzy.com

Abstract

This paper describes the joint participation of Inria Défense et Sécurité and Storyzy to the ASVspoof5 challenge. We participated in the closed conditions of the audio deepfake detection and of the spoofing-aware speaker verification tracks with the goal of evaluating the performance of countermeasures with a fixed set of training attacks. The proposed countermeasure system is the combination of three models with different architectures and training algorithms, including the exploration of a self-supervised learning pretraining approach. Specific data augmentation strategies are introduced to increase robustness to data transmission and generalization to unknown attacks. The submitted system achieves a minDCF of 0.297 for track1 and a min a-DCF of 0.295 for track2. It has a very small calibration error (actDCF of 0.298) despite the presence of unknown codecs and adversarial attacks within the evaluation corpus.

1. Introduction

The ASVspoof series of challenges [1] fosters the development of audio deepfake detection systems. The poor generalization of current countermeasures to in-the-wild data [2] highlights the need for more realistic datasets. The ASVspoof 2021 edition marked a significant advancement towards simulating more realistic scenarios, including the use of low-quality recordings and codec compression [3]. In the latest ASVspoof5 edition, the challenge has been elevated further by introducing more demanding conditions, such as crowdsourced adversarial attacks [4].

The main objective of an audio deepfake detection task is a *generalised countermeasure*, that is *spoofing detection solutions which perform reliably in the face of utterances generated with new or previously unseen spoofing attack algorithms and methods* [1]. We took part in the two tracks of the closed condition (audio deepfake detection and spoofing-aware automatic speaker verification) in order to evaluate the generalization of different models with a fixed gap between the training and evaluation attacks, independently of the problem of collecting a representative dataset of existing attacks.

1.1. Motivation

An ideal deepfake detection system would process as an anomaly detection mechanism, broadly applicable to imagery [5, 6]. Such a system would enhance generalization to novel attacks by modeling only the normal behavior, which in this context is from bonafide speech. Any spoofed utterances would then be identified as outliers by the system.

Recent studies [7] suggest that current countermeasure systems do not sufficiently model the bonafide class, whereas self-supervised learning (SSL) front-ends can be effectively used to design countermeasures [8]. Motivated by this claim, we investigated using SSL for deepfake detection.

In the solution we propose, this SSL detection approach is supplemented by fully supervised approaches based on well established architectures of the domain. Integrating the potential strengths of the different approaches, the final decision is made by combining the scores using a fusion technique.

1.2. Combination of systems

A combination of several different systems is expected to be more robust to new attacks or conditions thanks to the complementarity of the sub-systems. As a consequence, we explored three families of models for audio deepfake detection.

Two families of models were trained end-to-end for binary classification. ResNet models are well-known speakers verification architectures [9] and have been successfully applied to the audio deepfake detection task [10, 11]. AASIST [12] is an end-to-end architecture which constituted one of the baselines of the ASVspoof5 edition.

The third system is an SSL-based approach. We chose the Vision Transormer Masked AutoEncoder (Vit-MAE) [13, 14] approach, which has been shown to work efficiently on small datasets such as CIFAR10 [15]. After the SSL pretraining phase, a binary classifier is trained for audio deepfake detection.

The combination of these systems is then calibrated with a simple logistic regression [16], achieving a very small calibration error on the evaluation corpus for the target operating point of track1. In order to actually evaluate the effectiveness of this system as a CM for SASV, we also submitted it to the track2 closed condition. We thus trained an ASV system abiding by the rules of the challenge. The score fusion of CM and ASV was performed with the recently introduced non linear fusion method [17], with the script provided by the organizers of the challenge.

1.3. Data augmentations

Data augmentations are a key component of the training of audio deepfake detection systems [3]. In addition to standard speech data augmentations, we adopted a two-fold data augmentation strategy with the twin goals of increasing robustness to numerical transmission degradations and of improving generalization to unknown attacks.

In the context of numerical transmissions, an audio signal

10.21437/ASVspoof.2024-3

Table 1: *Subsets of the asvspoof5 corpus.*

Dataset	Usage	# speakers	# bonafide utterances	# spoofed utterances	# trials	attacks
asvspoof5-train-train	training	327	15156	130440	-	A01-A08
asvspoof5-train-dev	validation	73	3613	32858	-	A01-A08
asvspoof5-dev-cal	calibration	392	15527	53560	138514	A09-A16
asvspoof5-dev-eval	evaluation	393	15807	56056	143942	A09-A16

can be altered in several ways. Codecs are an essential function in networks, with the objective of adapting the signal to the available bandwidth, which results in an alteration of the audio signal. Their effects can accumulate depending on the networks they pass through. To increase robustness to numeral transmissions, we explored a set of data augmentations with different codecs [18]. Based on the performance on the progress set, we picked a set of various codecs with moderate degradation of the audio. In spite of a competitive pooled performance, our submission on the evaluation corpus suffers from a significative drop in performance for several codecs. This suggests that more aggressive codec augmentations could have performed better, maybe with the need of adapting the training recipe of the models.

Beyond their performance evaluated on a specific dataset, countermeasure systems are also sensitive to the performance of the attack systems intended to circumvent them. During system development, early experiments with the ASVspoof5 development dataset showed a drop in performance due to a concatenation-based attack. From human perception, this attack sounds different from the attacks of the training set and illustrates how difficult it is to generalize to new attacks. For this class of attacks, we decided to design a specific data augmentation strategy to enhance the global performance of our detection system.

1.4. Organization of the paper

Section 2 presents data usage and data augmentation techniques, our two main contributions in this part being codec augmentation and concatenation-based augmentation. Section 3 describes our implementation of the CM systems. Section 4 describes our ASV system. Experimental results are presented in Section 5 and discussed in Section 6.

2. Data

In this section, we describe how the available data has been used. Indeed, we took part in the closed condition of the challenge where the data is restricted to the ASVspoof5 corpus, and VoxCeleb2 for track2. Notably, we leveraged three kinds of data augmentation: standard speech processing augmentations, codec augmentations and a concatenation-based augmentation.

2.1. Dataset usage

One CM system was trained for track1 using only the ASVspoof5 corpus. It was also used for track2, in combination with an ASV system that was trained on VoxCeleb2 [19]. The split of the ASVspoof5 corpus is presented in Table 1.

The ASVspoof5 corpus contains a train and a dev subset. Each set is composed of eight attacks. We split the `asvspoof5-train` corpus into two subsets: `asvspoof5-train-dev` containing 20% of the speakers and used for validation, and `asvspoof5-train-train` with the remaining ones.

The CM systems were calibrated on unknown attacks to be more representative of the task. To this end, we split the `asvspoof5-dev` corpus into two subsets: `asvspoof5-dev-cal` used for calibration and `asvspoof5-dev-eval` used for evaluation on unknown attacks. Since these datasets were also used for track2, the split was based on the `asvspoof5-dev` trial list for track2, with the constraint that speakers belonging to a same trial should belong to the same subet. There is exactly one split with this constraint and this split is perfectly balanced, with 392 and 393 speakers within each subset.

2.2. Data augmentations

2.2.1. Standard data augmentations

All CM and ASV systems were trained with a subset of the following standard speech data augmentations:

- additive noise with artificial white noise and MUSAN [20] (noise subset only), with an SNR between 0 and 15 dB ;
- reverberation with both artficial and real room impulse response (RIR corpus) ;
- speed perturbation ($\times 0.9$ or $\times 1.1$) ;
- pitch perturbation (between -400 and $+400$ cent) ;
- specaugment [21].

2.2.2. Codec data augmentations

One of the main objectives of the ASVspoof5 challenge is to test the robustness of the systems to different acoustic codecs. Accordingly, we designed an augmentation strategy which consists of passing a signal through the encoding and decoding processes of a codec [18].

Two lists of codecs have been tested: `light-codec-list` and `full-codec-list`. The

Table 2: *Codec data augmentations. The first two columns show which codecs are included in* `full-codec-list` *and* `light-codec-list`.

`full`	`light`	Codec	Bit rate range (kb/s)
x	x	AAC	20.0, 60.0
x		AAC	10.0
x	x	Opus	2.0
x	x	AMR wide band	14.25, 32.05
x		AMR wide band	6.60, 14.25
x		AMR narrow band	4.75, 6.70, 12.20
x	x	GSM	
x		VOIP	15.0, 20.0
x		CODEC2	0.45, 1.4, 3.2

Table 3: *Audio deepfake detection performance on development sets. The submitted CM system is system 9.*

N°	Model	Data augmentations	dev-eval EER	no-A12 EER	only-A12 EER	progress minDCF	progress actDCF	progress Cllr	progress EER
1	AASIST	(*baseline*)	15.38	7.80	73.21	0.4451	0.4781	0.6573	17.13
2		`light-codec` + `B12`	4.60	4.83	2.53	-	-	-	-
3	ResNet	`nocodec`	11.78	1.52	51.0	0.1818	0.4102	0.5817	7.37
4		`light-codec`	11.37	1.31	54.8	0.1319	0.1450	0.2725	5.14
5		`full-codec`	12.03	3.90	42.95	0.2143	0.2147	0.2838	8.13
6		`light-codec` + `B12` (*cal on aug*)	0.96	0.86	1.56	0.0783	*0.1837*	*0.2194*	2.98
7	ViT-MAE	`full-codec`	13.52	5.30	59.22	0.3889	0.4650	0.6146	17.44
8	fusion	6 + 2	0.89	0.85	1.18	0.0767	0.0794	0.1165	2.96
9		6 + 2 + 7	0.74	0.64	1.49	0.0764	0.0848	0.1177	2.93

second is an extension of the first including more degradations, particularly in terms of acoustic bandwidth reductions. Codec details and repartition in the two lists are given Table 2.

2.2.3. Concatenation-based augmentation

Guided by the poor performance of our models on the A12 attack, we designed another augmentation first to tackle this kind of attacks and to further improve the models. A quick inspection of this aforementioned attack suggested a different spoofing strategy preserving local speech naturalness.

To mimic this attack, we defined a process for modifying a natural audio signal. The signal is divided into segments with random durations following a normal distribution with mean μ and variance σ^2. Each sequence is then modified by a random permutation. Finally, the transformed signal is simply composed by concatenating the elementary signals from each segment. This transformation, which we named `B12`, was applied offline and should be considered more as an attack used for training detection models rather than a strict augmentation applied on the fly. We excluded from this process the audio segments whose mean amplitudes are below $5e^{-2}$ on a normalized scale between [-1.0, 1.0]. Two sets of parameters were used to define two different kinds of transformations closed enough to A12: ($\mu = 8e^{-2}$, $\sigma^2 = 2.5e^{-5}$) and ($\mu = 0.2$, $\sigma^2 = 2.5e^{-5}$)

3. Countermeasure systems

Three kinds of countermeasure (CM) models were considered. First, we trained a ResNet model, replicating a speaker identification recipe for the audio deepfake detection task. We also trained the AASIST model [12], an end-to-end architecture for audio deepfake detection. Finally, our focus shifted toward Self-Supervised Learning and a ViT-MAE [13, 14] was pretrained on the training data of the challenge. The model was then finetuned for binary classification. This strategy is a first step towards an anomaly detection system not relying on binary classification, which has been shown to generalize poorly [2, 22].

3.1. ResNet

This model is an adaptation of a VoxCeleb speaker identification recipe with ResNet34 model [9]. ResNet models have been successfully applied to the audio deepfake detection task [10, 11].

In our implementation, the input features are log-melspectrograms extracted using 40 filterbanks, 25ms window and 10ms shift durations. The model has a ResNet34 architecture with 2d convolutions. It is followed by an attentive statistical pooling layer (mean and standard deviation) and a classification head. It has 15.8M parameters. It is trained for binary classification, with the cross-entropy loss. The model is trained on balanced minibatches of size 32 with segments of fixed duration (3s). We use the Adam optimizer with a fixed learning rate of 10^{-3}. The model with best validation loss is selected.

3.2. AASIST

AASIST [12] is a state-of-the-art architecture for audio deepfake detection. Our implementation is very similar to the baseline model provided by the challenge organizers. The main difference is that we do not restrict inference to the first four seconds of each utterance but run inference on the total duration of each utterance.

The model takes raw waveforms as input and is trained for binary classification to minimize the cross-entropy loss. We train it with the Adam optimizer with a learning rate of 10^{-4} and a cosine annealing scheduler. We use balanced minibatches of size 24 and of segments of fixed duration (3s). The model with best validation loss is selected.

3.3. ViT-MAE

The amount of data of the challenge being limited, we had to rely on a pretraining strategy that could work in the low-data regime. We chose the ViT-MAE [13, 14] approach, which has been shown to work on small datasets such as CIFAR10 [15]

This model takes as input the log-melspectrogram of an input waveform of 500 ms duration. The mels are extracted using 128 filterbanks, a 32 ms window length and a 8 ms shift duration. The ViT-MAE consists in training a ViT as a masked variational autoencoder. Its input is split into non-overlapping patches among which 75% are discarded. The task of the autoencoder is to reconstruct the missing patches based on the remaining ones.

To circumvent the difficulties naturally arising when training ViTs [23, 24], we modified the architecture to use LayerScale [25] and discarded the learning of the normalization layers [26]. We also carefully initialized the network to ensure its output is standardized when its input is. Lastly, we used the ConvViT approach [24] to further ensure the stability and reproducibility of trainings.

The model pretrained from scratch using SSL is the ConvViT-base [24]. Then, a three-layer MLP with a hidden dimension of 512 was trained on top of the frozen encoder to classify segments of 500 ms. The utterance-level prediction was computed using the average of the logposteriors of all non-overlapping segments of 500 ms.

The pretraining was performed over 800 epochs using the

AdamW optimizer [27] with a learning rate of 8×10^{-4} and a weight decay of 5×10^{-2}. We used linear warmup [28] for 5% of the steps and cosine annealing for the rest of the training. The model was pretrained with an effective minibatch size of 16000 segments of 500 ms without data augmentation on both bonafide and spoofed utterances. The training of the MLP followed the same recipe except that the model was trained for 50 epochs, the loss was the cross-entropy and data augmentation was used, including the codec augmentation using the `full-codec-list`.

3.4. Fusion and calibration

All the three models' final layers are trained using the cross-entropy loss. We assimilate the logit activations of the output layer as loglikelihood values for the hypotheses bonafide and spoofed. The resulting LLRs are then calibrated with a logistic regression[16] which is trained on the `asvspoof5-dev-cal` dataset. The same logistic regression model is used for score-level system fusion.

4. Spoofing-aware automatic speaker verification

4.1. Automatic Speaker Verification system

Abiding by the rules of the track2 closed condition, we trained an Automatic Speaker Verification system on the VoxCeleb2 corpus [19], without the MUSAN speech and music subsets for data augmentation. It is trained without the codec data augmentations. The model is a r-vector system from the Wespeaker recipe on VoxCeleb2 [29].

The input features are log-melspectrograms extracted using 80 filterbanks, 25ms window and 10ms shift durations. Cepstral Mean and Variance Normalization (CMVN) is applied. We use a ResNet34 architecture [9] with a temporal statistical pooling layer (mean and standard deviation). The model is trained for 150 epochs with an Arc-margin objective [30].

Embeddings of dimension 256 were extracted from the trained ResNet model and were centered with the mean vector obtained from the `asvspoof5-train-dev` corpus. Then, scoring was performed with cosine similarity between test and enrollment embeddings. Verification scores were normalized with adaptive symmetric score normalization (top n=300) with a cohort extracted from VoxCeleb2 [31]. Finally, the speaker verification LLRs were calibrated with a logistic regression trained on target and non target trials of the subset `asvspoof5-dev-cal`, excluding spoofed trials.

4.2. ASV and CM score fusion

Non linear fusion of the CM and ASV scores was performed with the score fusion script provided by the challenge organizers [17] to produce Spoofing-aware Automatic Speaker Verification scores. The distribution of the CM and ASV scores was learned on the `asvspoof5-dev-cal` subet of the dev trial list.

5. Experimental results

Performances of the countermeasure models on the `asvspoof5-dev-eval` and `progress` datasets are reported in Table 3.

For easier reading we only report EER on the `asvspoof5-dev-eval` corpus but report all challenge metrics on the progress set. The DCF is computed with an operating point given by the challenge, $\pi_{spf} = 0.05$, $C_{miss} = 1$, $C_{fa} = 10$, and is normalized [1]. All systems are calibrated on the `asvspoof5-dev-cal` dataset, with the exception of system 4 which is calibrated on an augmented version of the dataset and which suffers from a high calibration error on the progress set. System 2 has not been evaluated on the progress set during the progress phase of the challenge.

The models have also been evaluated on the `asvspoof5-train-dev` corpus. All models exhibit an EER below 1 % on this corpus, confirming that detection of known attacks is an easy task.

5.1. Effect of data augmentation on countermeasure systems

Data augmentation techniques have been selected based on experiments with the ResNet architecture. We observe that the different versions of codec augmentation (`nocodec`, `light-codec-list` and `full-codec-list`) do not have a significant impact on performance on the `asvspoof5-dev-eval` corpus. The impact is different on the `progress` set where the `light-codec-list` augmentation achieves a significant improvement over `nocodec`. We assume that the low performance of the `full-codec-list` augmentation on the progress set may be explained by the bad convergence of the model. Better generalization of a model with `full-codec-list` augmentation may be achieved with a careful tuning of the optimization hyperparameters.

The codec augmentations during the model training phase, nonetheless, improves calibration of the model. The difference between actDCF and minDCF for the `nocodec` model (0.4102 vs 0.1818) is reduced for the `light-codec-list` (0.1450 vs 0.1319) and `full-codec-list` (0.2147 vs 0.2143). It is important to note that data augmentation was applied exclusively to the training set of the model and not to the calibration dataset.

During the system development phase, we performed per-attack evaluation of the systems. We noticed that the high error rates on the `asvspoof5-dev-eval` corpus was mainly due to the A12 attack where the systems 1, 3, 4 and 7 achieve worst than random performance. To illustrate this effect, we provide performance metrics separately for the A12 attack and for the set of all other attacks excluding A12. We observe that the introduction of the `B12` augmentation (system 6) drastically improves performance on the A12 attack.

The best combination of augmentations, `light-codec-list` augmentation and `B12` augmentation, has been selected to train the ResNet model. Applied to the AASIST architecture, it achieves a significant improvement (comparison of systems 1 and 2). Because of time constraints

Table 4: *Automatic Speaker Verification Equal Error Rate (%) on VoxCeleb1 and ASVspoof5 development sets with and without speech used in data augmentation. Only the ASV system trained without MUSAN speech and music is used for scoring.*

	vox1-O clean	vox1-E clean	vox1-H clean	asvspoof5-dev (no spoof)
w/ MUSAN speech & music	0.787	0.964	1.726	-
w/o MUSAN speech & music	0.851	0.990	1.787	1.285

Table 5: *Performance on progress and on dev track2. The submitted system corresponds to the last row.*

CM model	ASV model	asvspoof5-dev-eval			progress		
		min a-DCF	min t-DCF	t-EER	min a-DCF	min t-DCF	t-EER
6	ResNet	0.0221	0.1070	0.90	0.0986	0.1826	4.11
9		0.0210	0.1005	0.85	0.1006	0.1859	4.17

during the challenge, only the `full-codec-list` augmentation has been applied to the ViT-MAE model (system 7) for the binary classification phase (not pretraining). The submitted countermeasure system for both tracks is the system 9, fusion of systems 6 (resnet with `light-codec-list` and `B12` augmentations), 2 (AASIST with `light-codec-list` and `B12` augmentations) and 7 (VITMAE with `full-codec-list` augmentation).

5.2. Spoofing-aware automatic speaker verification

Table 4 reports indicative performance of the ASV system on VoxCeleb1 when the model is trained with and without MUSAN speech and music splits for data augmentation. As expected, when MUSAN speech and music splits are used, the model performs better than when they are not used (0.787 vs 0.851 on `vox1-O-clean`). However, the increase in performance is modest, suggesting that the ResNet34 does not heavily depend on MUSAN speech and music splits to achieve proper performance. On the `asvspoof5-dev` set, with spoofed trials removed from scoring (classic EER with target and non-target pairs), the ASV model performs well with an EER of 1.285 %, indicating that the speaker verification task is relatively easy on the dev set.

In Table 5, we evaluate the SASV systems constituted of the combination of the ASV system with two CM systems: the best single system (system 6) and the fusion of ResNet, AASIST and ViT-MAE (system 9). For consistency between tracks 1 and 2, we submitted the combination with the fusion system, even though its performance is slightly worse on the progress dataset.

5.3. Performance on the evaluation corpus

In Tables 6 and 7, we report the performance of the submitted systems on the evaluation corpus for tracks 1 and 2 closed condition. This level of performance looks reasonable but shows a drop in comparison with the dev and progress corpora. We notice that the system is remarkably well calibrated for the target operating point of track1, with a very small gap between actDCF and minDCF.

The evaluation dataset is described in [4] where our submission is referred as T24. In Figures 1 and 2, we plot the actDCF of the submitted system for track1 for each attack and condition, and the min a-DCF for track2. First, we observe that the system performs well in the absence of codec. The pooled actDCF result with nocodec (0.186) is only twice the value on the progress set (0.085) despite the presence of unknown adversarial attacks.

There is no catastrophic behavior comparable to the A12 attack observed in the dev corpus. Additionally, the normalized actDCF remains below 1.0 across all attacks and conditions (not worst than a default system). For the nocodec condition, the system achieves an actDCF below 0.05 for eight of the sixteen evaluation attacks. The adversarial attacks such as Malafide [32] (A18 and A20) and Malacopula [33] (A27, A30, A31, A32) are very effective on the submitted system and are responsible of the majority of errors on the nocodec condition

On the contrary, the system suffers from a significative drop in performance for most codecs, until a three-fold increase of the actDCF for codec-7 compared to the nocodec condition. 8kHz bandwidth conditions (codec-8, codec-9, codec-10, codec-11) are challenging but the worst actDCF values are obtained when the Encodec [34] neural audio compression is applied (codec-4 and codec-7).

6. Discussion

6.1. Generalization to unknown attacks

The generalization to *a priori* unknown attacks is the main challenge of deepfake detection. The difficulty to detect the A12 attack of the development corpus is a perfect illustration of this problem. Our solution was to introduce a specific concatenation-based data augmentation, which basically solves this problem on the development corpus and seems useful on the progress dataset. Fortunately, the proposed countermeasure systems generalize quite well to the attacks of the evaluation corpus, with no catastrophic behavior comparable with attack A12.

But we had no guarantee that this approach would generalize. The proposed solution was to add augmented spoofed utterances more representative of a class of new target attacks to the training set. This approach exposes the practitioner to a potential catastrophic behavior when exposed to new attacks being very different from the attacks of the training set. We believe that a simple binary classification approach is insufficient to establish the level of trust required for effectively handling new attacks in practical applications. We plan to explore additional anomaly detection methods that focus on a more detailed modeling of bonafide speech. This was the main motivation of our work on the ViT-MAE system presented in subsection 3.3, even though we were unable to implement an efficient anomaly detection CM within the challenge timeline.

6.2. Robustness to codec degradation

Robustness to codec degradation can be partially achieved with codec data augmentation of the training set. This approach achieved significant gains on the development and progress sets, and may be responsible of the competitive performance of the submitted system on the evaluation corpus.

We selected a set of codec augmentations with moderate degradations, most of them do not operate a frequency band reduction, which have been successfully applied to the CM systems without the need to modify the optimization hyperparameters. The observed performance drop on the evaluation set for various codecs suggests that more aggressive codec data augmentation methods could have been necessary. Even though these kind of augmentations were explored during the development phase of our solution, we did not select them due to their lower performance on nocodec conditions. Their effective ap-

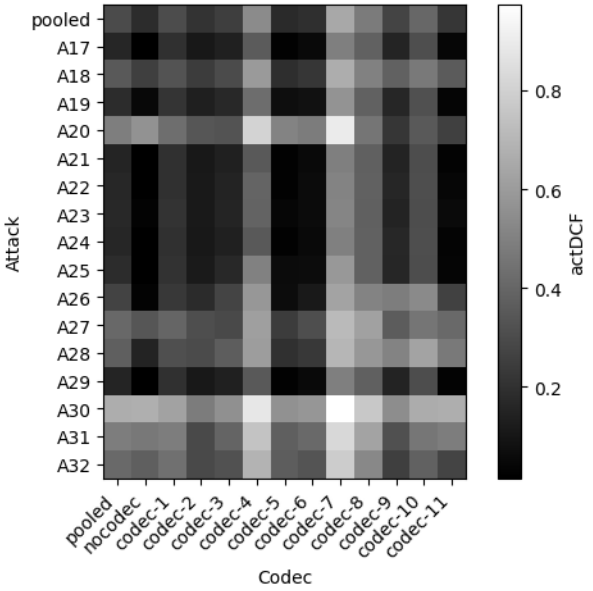

Figure 1: *actDCF of the submitted system on track1 evaluation corpus. Attacks and codecs are described in [4].*

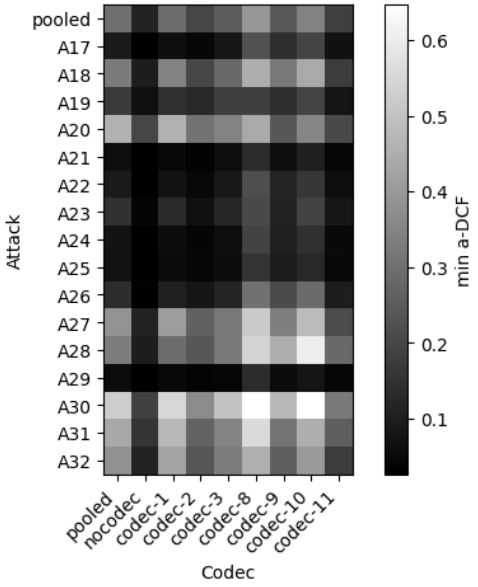

Figure 2: *min a-DCF of the submitted system on track2 evaluation corpus.*

plication to the CM systems seems reachable but will probably need an adaptation of the optimization hyperparameters. Moreover an increased performance for spoofing-aware speaker verification can be expected from the application of codec specific data augmentations on the speaker verification system.

6.3. Calibration

Practical use of a countermeasure system requires scores to be calibrated. For the first time the ASVspoof5 challenge encouraged participants to submit calibrated LLRs [4]. We calibrated the system with a simple logistic regression and obtained a very low calibration error on the evaluation corpus. However, there is an inherent difficulty in calibrating a countermeasure system, due firstly to unknown attacks, but also to ensemble-based architectures, where different subsystems may be trained on different subsets of attacks and react differently depending on the attack. We hope that there will be more interest in the calibration of countermeasure systems, and we plan to study this issue in greater depth over the coming months.

Table 6: *Performance on the evaluation dataset of the submitted system for track1*

minDCF	actDCF	Cllr	EER
0.297	0.298	0.418	10.43

Table 7: *Performance on the evaluation dataset of the submitted system for track2*

min a-DCF	min t-DCF	t-EER
0.295	0.618	9.58

7. Conclusion

This paper describes the joint submission of Inria Défense et Sécurité and Storyzy for the ASVspoof5 challenge closed condition. The submitted countermeasure system is the combination of three models trained for binary classification: ResNet, AASIST and ViT-MAE. The ViT-MAE model is pretrained with a masked autoencoder objective. Data augmentations are applied to enforce robustness to data transmission and to improve performance on concatenation attacks. For track2, the system is combined with a custom automatic speaker verification system, trained on VoxCeleb2 without MUSAN speech and music subsets to abide by the rules of the challenge. The submitted system achieves a competitive performance on the evaluation corpus of both tracks, which is constituted of challenging adversarial attacks and codec conditions. It is particularly well calibrated for the target operating point of track1.

8. Acknowledgements

This work was performed using HPC resources from GENCI-IDRIS (Grant AD011014982).

9. References

[1] Hector Delgado, Nicholas Evans, Jeeweon Jung, Tomi Kinnunen, Ivan Kukanov, Kong Aik Lee, Xuechen Liu, Hye-jin Shim, Hemlata Tak, Massimiliano Todisco, Xin Wang, and Junichi Yamagishi, "ASVspoof 5 evaluation plan (phase 2)," online, accessed 31-July-2024.

[2] Nicolas M. Müller, Pavel Czempin, Franziska Dieckmann, Adam Froghyar, and Konstantin Böttinger, "Does audio deepfake detection generalize?," in *Proc. Interspeech*, 2022, pp. 2783–2787.

[3] Xuechen Liu, Xin Wang, Md Sahidullah, Jose Patino, Héctor 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," in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023, vol. 31, pp. 2507–2522.

[4] 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, and Junichi Yamagishi, "ASVspoof 5: Crowdsourced speech data, deepfakes, and adversarial attacks at scale," in *ASVspoof Workshop 2024 (accepted)*, 2024.

[5] Jianwei Fei, Yunshu Dai, Peipeng Yu, Tianrun Shen, Zhihua Xia, and Jian Weng, "Learning second order local anomaly for general face forgery detection," in *Proc. IEEE/CVF CVPR*, 2022, pp. 20238–20248.

[6] Chao Feng, Ziyang Chen, and Andrew Owens, "Self-supervised video forensics by audio-visual anomaly detection," in *Proc. IEEE/CVF CVPR*, 2023, pp. 10491–10503.

[7] Hye-jin Shim, Md Sahidullah, Jee-weon Jung, Shinji Watanabe, and Tomi Kinnunen, "Beyond silence: Bias analysis through loss and asymmetric approach in audio anti-spoofing," arXiv preprint arXiv:2406.17246, 2024.

[8] Xin Wang and Junichi Yamagishi, "Investigating self-supervised front ends for speech spoofing countermeasures," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2022, pp. 100–106.

[9] Pierre-Michel Bousquet, Mickael Rouvier, and Jean-Francois Bonastre, "Reliability criterion based on learning-phase entropy for speaker recognition with neural network," in *Proc. Interspeech*, 2022, pp. 281–285.

[10] Weicheng Cai, Haiwei Wu, Danwei Cai, and Ming Li, "The DKU replay detection system for the ASVspoof 2019 challenge: On data augmentation, feature representation, classification, and fusion," in *Proc. Interspeech*, 2019, pp. 1023–1027.

[11] Tianxiang Chen, Elie Khoury, Kedar Phatak, and Ganesh Sivaraman, "Pindrop Labs' submission to the ASVspoof 2021 challenge," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, pp. 89–93.

[12] 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. IEEE ICASSP*, 2022, pp. 6367–6371.

[13] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF CVPR*, 2022, pp. 15979–15988.

[14] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino, "Masked spectrogram modeling using masked autoencoders for learning general-purpose audio representation," in *Volume 166: Holistic Evaluation of Audio Representations*. 2022, PMLR.

[15] Kentaro Yoshioka, "https://github.com/kentaroy47/vision-transformers-cifar10," online, accessed 31-July-2024.

[16] Luciana Ferrer, "https://github.com/luferrer/calibrationtutorial," online, accessed 31-July-2024.

[17] 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 (accepted)*, 2024.

[18] Rohan Kumar Das, "Known-unknown data augmentation strategies for detection of logical access, physical access and speech deepfake attacks: ASVspoof 2021," in *2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 29–36.

[19] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[20] David Snyder, Guoguo Chen, and Daniel Povey, "MUSAN: A music, speech, and noise corpus," 2015, arXiv, arXiv:1510.08484.

[21] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[22] Davide Alessandro Coccomini, Roberto Caldelli, Fabrizio Falchi, and Claudio Gennaro, "On the generalization of deep learning models in video deepfake detection," in *Journal of Imaging*. 2023, vol. 9, p. 89, Multidisciplinary Digital Publishing Institute.

[23] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou, "Training data-efficient image transformers & distillation through attention," in *Proc. ICML*. 2021, pp. 10347–10357, PMLR.

[24] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, and Ross Girshick, "Early convolutions help transformers see better," in *Advances in neural information processing systems*, 2021, pp. 30392–30400.

[25] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Herve Jegou, "Going deeper with image transformers," in *IEEE/CVF ICCV*, 2021, pp. 32–42.

[26] Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin, "Understanding and improving layer normalization," in *Advances in neural information processing systems*, 2019, pp. 4381–4391.

[27] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," in *Proc. ICLR*, 2019.

[28] Jerry Ma and Denis Yarats, "On the adequacy of untuned warmup for adaptive optimization," in *Proc. AAAI Conference on Artificial Intelligence*, vol. 35, pp. 8828–8836.

[29] Shuai Wang, Chengdong Liang, Xu Xiang, Bing Han, Zhengyang Chen, Hongji Wang, and Wen Ding, "Wespeaker baselines for VoxSRC 2023," 2023, arXiv:2306.15161.

[30] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF CVPR*, 2019, pp. 4690–4699.

[31] Pavel Matějka, Ondřej Novotný, Oldřich Plchot, Lukáš Burget, Mireia Diez Sánchez, and Jan Černocký, "Analysis of score normalization in multilingual speaker recognition," in *Proc. Interspeech*, 2017, pp. 1567–1571.

[32] 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, number arXiv:2306.07655, pp. 2868–2872.

[33] 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. ASVspoof5 workshop (submitted)*, 2024.

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