Towards generalisable and calibrated audio deepfake detection with self-supervised representations

Octavian Pascu[1], Adriana Stan[1,3], Dan Oneata[1], Elisabeta Oneata[2], Horia Cucu[1]

[1]POLITEHNICA Bucharest, Romania   [2]Bitdefender, Romania
[3]Technical University of Cluj-Napoca, Romania

octavian.pascu@upb.ro   adriana.stan@com.utcluj.ro   dan.oneata@gmail.com
eoneata@bitdefender.com   horia.cucu@upb.ro

Abstract

Generalisation—the ability of a model to perform well on unseen data—is crucial for building reliable deepfake detectors. However, recent studies have shown that the current audio deepfake models fall short of this desideratum. In this work we investigate the potential of pretrained self-supervised representations in building general and calibrated audio deepfake detection models. We show that large frozen representations coupled with a simple logistic regression classifier are extremely effective in achieving strong generalisation capabilities: compared to the RawNet2 model, this approach reduces the equal error rate from 30.9% to 8.8% on a benchmark of eight deepfake datasets, while learning less than 2k parameters. Moreover, the proposed method produces considerably more reliable predictions compared to previous approaches making it more suitable for realistic use.

Index Terms: Deepfake detection, anti-spoofing, pretrained representations, generalisation, calibration, reliability estimation.

1. Introduction

The ability to synthetically generate audio data is constantly improving [1, 2, 3]. While these advancements have many beneficial applications (such as allowing speech-impaired persons to recover their voices or creating digital art and entertainment content), they can also serve malicious purposes (*e.g.*, cloning officials' voices to spread misinformation). Synthetic speech detection (or audio deepfake detection) attempts to prevent such misuses of the technology by developing methods that automatically estimate whether a given audio is real (bonafide) or fake (spoofed). While there is a sustained ongoing effort on this task [4, 5, 6], we argue that for the ensuing methods to be effective, they should strive for two properties: they should generalise and be trustworthy (*i.e.*, well calibrated).

Generalisable detection methods. Since synthesis methods are continuously evolving, it is unreasonable to expect that we will have access to training data similar to that encountered in practice. *Generalisation* is the capability of a model to perform well on data not seen at training time. However, Müller *et al.* [7] have shown that the generalisation abilities of popular audio deepfake detectors have been overestimated. They evaluate twelve top-performing detection models and show that none of them are able to generalise on an out-of-distribution dataset.

A possible explanation for the poor generalisation performance is related to the preprocessing peculiarities exhibited by the training dataset (ASVspoof'19 [8])—the silence duration [9] and the bitrate information [10] correlate with the ground truth. Given that the best deepfake detection models are high-capacity, they can easily learn such low-level spurious features. In this paper, we take an under-explored path and assess the power of strong pretrained representations to improve the generalisation capabilities of audio deepfake detectors. Our approach is motivated by the strong generalisation results shown in [11] in the context of image deepfake detection.

While self-supervised representations were also applied to the detection of spoofed speech, previous work only used smaller representations with modest results [4, 12]. Moreover, these approaches did not focus on generalisation and were tested only on the ASVspoof'19 dataset, and not other challenging datasets, such as In-the-Wild [7]. Other works finetuned these features or integrated them in more complex systems [4, 13], but in this way they lose the implicit generalisation power.

Well-calibrated detection methods. Deepfake detectors can be used to make critical decisions, so it is crucial to be reliable and trustworthy. If a detector outputs a fakeness score of 0.7 for a series of inputs, then we should expect that in 70% of the cases the input is indeed fake. A classifier with this property is known as *well calibrated*. Current research in general machine learning addresses this aspect, but surprisingly little work discusses the calibration of deepfake detectors. Recent works [14, 15] have tackled a related problem of estimating the uncertainty (or conversely certainty) in a prediction. Both papers use a similar method: first train a deepfake detector, then train a second classifier (using a frozen representation extracted from the detector) to estimate if the predictions of the first are correct or not. Here, we investigate if it is feasible to use the more direct method of estimating the uncertainty from the output probabilities of the detector, by computing the entropy over the outputs.

Our contributions are as follows: (i) We propose a simple yet effective approach for detecting spoofed audio signals. This approach achieves state-of-the-art results and produces better calibrated outputs, suitable for reliability estimation. (ii) We conduct a systematic study over the impact of various factors including the choice of self-supervised representations, the back-end classifier and the amount of training data. (iii) We benchmark our approach on eight different datasets including partially-spoofed and multilingual data, laying the grounds for future research on general deepfake detection.

2. Methodology

The task of audio deepfake detection is to predict whether a given audio is real (bonafide) or fake (spoof). We investigate the approach of first extracting a pretrained representation using frozen self-supervised models and then training a binary classifier (logistic regression) on top of these representations. Logistic regression estimates probabilities, which we use for the task of uncertainty estimation. While these components are not novel, their application to this setting is. As we will see, this solu-

Dataset short name	Real data	Langs.	Partial	Systems	SR kHz	Real count	Fake count	Duration seconds
ASV [8]	VCTK	en	✗	10 TTS, 3 VC	16	7k	63k	3.1±2.9
ITW [7]	YouTube	en	✗	N/A	16	20k	12k	4.2±6.6
TIM [16]	VidTIMIT	en	✗	12 TTS	16	430	20k	3.1±2.3
FoR [17]	Arctic, LJSpeech, VoxForge, YouTube, TED Talks	en	✗	6 TTS	16	34k	34k	3.0±4.5
PS [18]	VCTK	en	✓	19 TTS or VC	16	7k	63k	3.4±3.5
ODSS [19]	VCTK, Hi-Fi TTS, HUI-ACG, SLR-ES	en, es, de	✗	2 TTS	16	11k	19k	3.1±4.1
MLAAD [20]	M-AILABS	many (23)	✗	52 TTS	22	20k	80k	7.6±10.1

Table 1: *Datasets from the evaluation benchmark used in this paper. We consider only the test splits of these datasets. The duration of the audio files is given as the average plus-minus two times standard deviation.*

tion offers strong generalisation and calibration performance, as opposed to even more elaborate models.

Pretrained representations. We investigate self-supervised representations stemming from the wav2vec 2.0 [21] method. We have chosen this family of models because it has proved strong transfer abilities [22] and comes in multiple variants, enabling us to assess the importance of various factors, such as model size or pretraining data. Wav2vec 2.0 was designed to perform unsupervised pretraining on raw audio data and as a result learns useful speech representations without the need of phonetic or linguistic annotations. It uses contrastive predictive coding [23] to capture high-level feature and contextual information. Several wav2vec 2.0 extensions were subsequently developed. The first class of models is XLS-R [24], which learns cross-lingual speech representations. This is achieved by adding a shared quantisation module over the feature encoder representations, producing multilingual speech units, and thus sharing acoustic representations across languages. This model comes in different sizes: 300M, 1B or 2B parameters. The second class of models is WavLM [25], which considered the task of speech denoising in addition to the masked audio prediction task in wav2vec. The model aims to produce high-level features targeted towards other non-ASR tasks. We use all these variants in our experiments.

Calibration and reliability estimation. A classifier is calibrated if its predictions match the accuracy obtained for that particular level of confidence. We apply the logistic regression classifier which uses the cross-entropy loss. The cross-entropy loss is a proper loss [26], which improves the calibration properties. This choice avoids the need of other post-processing techniques such as Platt's scaling [27]. Calibration is also related to generalisation: Carrell *et al.* [28] have shown that the calibration error is bounded by the generalisation error. This means that better calibrated classifiers are obtained by improving their generalisation. Calibrated probabilities help with related downstream tasks [29]. Here, we focus on reliability estimation, which is useful for rejecting examples for which the model is unsure. Given the estimated probability $\hat{y}$ of the audio being fake, we obtain uncertainty estimates by computing the entropy: $\hat{y} \log \hat{y} + (1 - \hat{y}) \log(1 - \hat{y})$. We use a unit-scaled variant of the entropy by dividing it by the maximum entropy $H_{\max}$, which is obtained for the uniform distribution, $y = 0.5$.

3. Experimental setup

Datasets. ASVspoof'19 (ASV) [8] is a popular deepfake detecion dataset, which we use for training in all our experiments. The fake audio samples are synthesised with 19 systems (6 systems in the `train` and `dev` splits, 13 in the `test` split). The

`train` and `dev` splits have 50k audio samples (5k real, 45k fake), while `test` has 70k samples (63k fake and 7k real). Similar to [30], we use both `train` and `dev` for training.

To test the out-of-domain generalisation capabilities we ensemble a benchmark of eight datasets, including partially-spoofed and multilingual datasets. Seven of these datasets are summarised in Table 1. Additionally, we use the `aug-dtw` version of TIM (denoted by TIM* hereafter), which has identical characteristics to the original dataset, except the audios underwent extra post-processing steps, such as time and pitch shifting, compression, filtering, and dynamic time warping.

Metrics. We evaluate the methods from two perspectives: (i) their discriminative power over fake and real samples; and (ii) their ability to produce calibrated predictions. These desiderata are measured by the *equal error rate* (EER) and *expected calibration error* (ECE) [31], respectively. In particular, ECE measures the absolute difference between the average predicted probability and the actual observed frequency within bins of predicted probabilities (we use 15 equally-spaced bins). The lower the ECE, the better the calibration.

Implementation details. We extract the self-supervised representations as the average of the last layer of hidden states. For classification, we use the `scikit-learn` [32] implementation of logistic regression with $C = 10^6$, implying a low regularisation coefficient, and set the maximum number of iterations to 1,000. Training on the full ASVspoof'19 dataset (50k samples) using 1920-dimensional features takes 40 seconds on a 64-core machine. Our code is available at: `https://github.com/danoneata/aletheia`.

4. Experimental results

4.1. How well do self-supervised representations generalise?

The most important quality of deepfake detectors is their generalisation capability. We measure this by training on the ASVspoof'19 dataset and evaluating on the benchmark described in Section 3. The results, which are shown in Table 2, contrast the performance of pretrained self-supervised representations (row 3) to that of the best models in the literature (row 1) and to the RawNet2 model (row 2), one of the best models on generalisation according to Müller *et al.* [7]. As self-supervised representation we use the 2B XLS-R variant from wav2vec 2.0, `wav2vec2/xls-r-2b`, which is the largest model and trained on most data; we analyze more variants in Section 4.3.

We observe that pretrained representations perform on average much better than RawNet2: 8.8% EER versus 30.9% EER. The performance is also better on each individual dataset with a single exception, TIM. Our approach also compares favourably

	Params.	Time		Memory	EER (%) ↓								
Method	*trainable*	train	pred	pred	ASV	ITW	TIM	TIM*	FoR	PS	OSDD	MLAAD	Mean
1 State of the art					**0.2** [33]	7.7 [34]	N/A	N/A	18.1 [35]	14.2 [18]	N/A	N/A	
2 RawNet2	25M	8h	0.03s	1.3GB	5.9±0.1	46.7±0.3	**2.4**±0.3	27.9±0.5	52.1±0.2	33.1±0.3	45.0±0.3	34.4±0.2	30.9
3 Ours	2k	4h	0.26s	9.3GB	0.5±0.1	**7.2**±0.3	11.5±1.1	**3.6**±1.4	**6.9**±0.2	**5.1**±0.9	**16.0**±0.4	**20.0**±0.3	**8.8**

Table 2: *Comparison in terms of EER with state-of-the-art on multiple out-of-domain datasets. We report mean and two times standard deviation using ten runs of bootstrap on the test data. On average, the proposed approach improves from 30.9% EER to 8.8% EER compared to RawNet2. We also show significant improvements when compared to available state-of-the art methods for each dataset.*
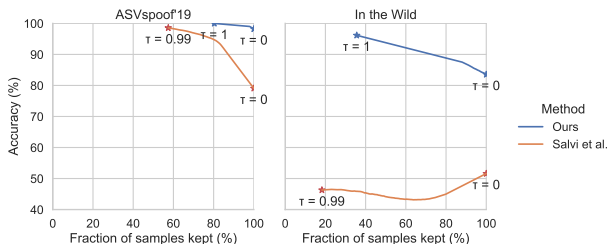

Figure 1: *Evaluation of reliability estimation in terms of accuracy and fraction of samples kept, as we vary the reliability threshold $\tau \in [0, 1]$. Our results (blue) are more reliable than those of Salvi et al. [15] (orange) on both metrics and datasets.*
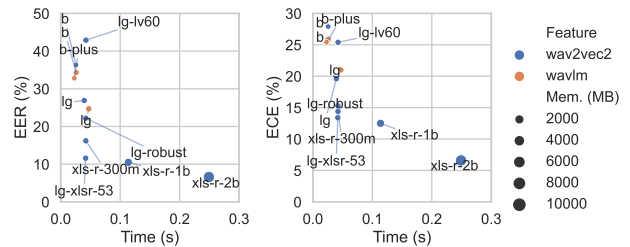

Figure 2: *Performance trade off as a function of inference time for the 11 variants of self-supervised representations. Marker area indicates peak video memory. Time and memory are averaged over 64 audio files, which average three seconds.*

to many of the state-of-the-art methods. These are much more complex methods, which are evaluated only on a handful of datasets. For several datasets we are the first to either report results (OSDD) or the first to report results in terms of EER (TIM, TIM*, MLAAD).

In terms of computational requirements, the cost of self-supervised representations is dominated by the feature extraction step. The time to process an audio of 3 seconds on a Tesla T4 GPU is around 0.3 s with a video memory consumption of around 9GB. These requirements are around an order of magnitude larger than those of RawNet2, but still reasonable in the absolute and attainable by commodity hardware.

4.2. How reliable are the self-supervised representations?

Another important characteristic of deepfake detectors is whether we can trust their predictions. Following Salvi *et al.* [15], we formulate this desideratum as the task of *reliability estimation*: we want the model to be able to assess the level of confidence in its predictions, a high confidence indicating a high probability that the prediction is correct. To this end, we encode the confidence in a prediction using the entropy of the generated probabilities (see Section 2): if the entropy is close to zero, the model deems the prediction to be reliable; conversely, high entropy indicates uncertain inputs. Salvi *et al.* [15] used a separate network to estimate the reliability which was trained on features from a pretrained deepfake audio model (in their case, RawNet2) to predict whether the predictions of the deepfake model are correct or not.

We use two metrics to measure the reliability estimation capabilities [36]: the fraction of data that is reliable and the classifier accuracy on this data. For our approach we produce curves by varying a threshold τ from 0 to 1 in steps of 0.01 on the unit-scaled entropy of each prediction. For the method of [15] we vary the threshold similarly, but on the maximum reliability score across segments. A threshold of 0.5 corresponds to their original evaluation: if all the audio's segments are unreliable (reliability

less than 0.5) then the entire audio is deemed unreliable.

We report results on the ASVspoof'19 and In-the-Wild datasets, but differently from [15], we evaluate on *all* the samples, not only fake samples. This is a more fair evaluation, since otherwise a classifier that predicts only fake labels with high confidence will obtain a perfect score. The results for [15] are computed over the predictions which were provided by the authors. The results are shown in Figure 1. We observe much better results than prior work on both datasets, in terms of both metrics, and at all thresholds. Naturally, there is a drop in performance when going out-of-domain (on In-the-Wild), but it is less severe than what we observe for the method of Salvi *et al.* [15]. Moreover, on In-the-Wild we see that we can trade off data kept for accuracy, which is not the case for the other method.

4.3. How do other self-supervised representations perform?

Pretrained self-supervised representations come in multiple variants, differing in terms of architecture, model size or pretraining data. Based on this information we decouple the performance on three axes: model family, model size, data. Table 3 shows the results for 11 variations of self-supervised representations belonging to two classses of models: wav2vec2 and wavlm. The results are given in terms of discrimination (EER) and calibration error (EER) on 4 out of the 8 datasets previously used.

We observe that both performance metrics improve on average with the size of the self-supervised model. When increasing the model size from 94M to 300M parameters, the mean EER decreases from 36.3% to 26.9% (rows 1–2) for wav2vec2 and from 32.8% to 24.7% (rows 10–11) for wavlm. The error further decreases from 16.2% to 6.6% (rows 6–8) when increasing the wav2vec2 model size from 300M to 2B parameters. The best mean ECE is also obtained for the largest 2B model: 6.6%.

Among the two families of representations, wav2vec2 appears to be the better representation for deepfake detection. While wavlm is only slightly better than wav2vec2

	Model	Pretraining data			EER (%) ↓					ECE (%) ↓				
Name	Size	Dur. (h)	Langs.	Datasets	ASV	ITW	TIM	FoR	Mean	ASV	ITW	TIM	FoR	Mean
`wav2vec2/`														
1 `base`	94M	1k	en	LS	4.0	45.9	59.4	36.2	36.3	5.1	61.3	6.9	38.5	27.9
2 `large`	300M	1k	en	LS	3.7	29.0	51.7	23.5	26.9	3.3	40.9	4.7	29.8	19.6
3 `large-lv60`	300M	53k	en	LL	2.5	50.3	90.3	28.6	42.9	2.2	61.1	2.77	35.6	25.4
4 `large-robust`	300M	65k	en	LS, LL, SF, CV/en, VP/en	2.9	21.3	52.3	12.3	22.2	4.1	31.8	16.2	9.4	15.3
5 `large-xlsr-53`	300M	56k	many	CV, BBL, MLS	1.1	25.0	**4.7**	15.8	11.6	2.3	36.7	**1.6**	13.3	13.4
6 `xls-r-300m`	300M	436k	many	CV, BBL, MLS, VP, VL	1.0	21.3	37.8	**5.1**	16.2	2.2	38.7	8.4	8.4	14.4
7 `xls-r-1b`	1B	436k	many	CV, BBL, MLS, VP, VL	1.3	18.7	13.2	9.0	10.5	3.1	26.9	8.4	11.9	12.5
8 `xls-r-2b`	2B	436k	many	CV, BBL, MLS, VP, VL	**0.6**	**7.2**	11.3	6.8	**6.5**	1.8	16.1	2.3	**6.3**	**6.6**
`wavlm/`														
9 `base`	94M	1k	en	LS	4.8	37.6	50.2	44.6	34.3	4.5	50.4	2.2	46.6	25.9
10 `base-plus`	94M	84k	en	LS, GS, VP	3.6	38.6	50.3	38.7	32.8	3.0	54.6	2.3	41.8	25.4
11 `large`	300M	84k	en	LS, GS, VP	1.8	32.8	55.6	8.6	24.7	1.5	55.8	2.9	23.8	21.0

Table 3: *Evaluation of self-supervised representations for deepfake detection. By keeping various dimensions fixed, we can asses the impact of model size (rows 1–2, 6–8, 10–11), pretraining data (rows 2–6, 9–10), and representation type (row 1 vs. row 9). The pretraining datasets are LibriSpeech (LS), LibriLight (LL), Switchboard & Fisher (SF), CommonVoice (CV), VoxPopuli (VP), Babel (BBL), Multilingual LibriSpeech (MLS), VoxLingua107 (VL), GigaSpeech (GS).*

at 94M parameters (34.3% EER, row 9, vs. 36.3% EER, row 1), `wav2vec2` has a much improved 300M-parameter model: 11.6% EER (row 5) vs. 24.7% EER (row 11). Similar observations can be made for the calibration performance.

The conclusions regarding the pretraining data are not as clear. Sometimes more diverse training data improves performance (row 4 vs. rows 2–3; row 10 vs row 9), but in other cases adding more data just hurts performance (row 6 vs. row 5). We find this result surprising and worth of further investigation.

In Figure 2, we also analyze the computational requirements of these representations. We observe that the best performing representation, `xls-r-2b`, has the largest costs in terms of both memory and time, albeit still reasonable in the absolute. In order to optimise different aspects, one may choose other representations, for example one with a good trade-off among the considered metrics is the `large-xlsr-53` variant.

4.4. How important is the classifier?

We have experimented with two more flexible models on top of the frozen `wav2vec2/xls-r-2b` representations: a three-layer multilayer perceptron (MLP) with ReLU activation, and a self-attention layer (SAL) followed by a linear layer. Additionally, we have investigated a stronger regularisation value for logistic regression, $C = 1$. Results are shown in Table 4. On average, logistic regression obtains best results on both metrics, although there are variations across the test datasets. Surprisingly, the less regularised variant, $C = 10^6$, generalises better. The reason is perhaps that the logistic model, being a linear model, is already highly constrained, so further regularisation ($C = 10^0$) results in underfitting and poorer results.

		EER (%) ↓					ECE (%) ↓				
Classifier	C	ASV	ITW	TIM	FoR	μ	ASV	ITW	TIM	FoR	μ
SelfAtt		1.3	9.3	26.59	8.1	11.3	1.5	**7.0**	33.2	18.7	15.1
MLP		**0.4**	10.3	17.2	**4.9**	8.2	**0.8**	20.7	4.2	8.0	8.4
LogReg	10^0	**0.4**	8.6	19.1	7.8	9.0	1.4	14.5	**2.3**	**6.3**	**6.1**
LogReg	10^6	0.6	**7.2**	**11.3**	6.8	**6.5**	1.8	16.1	**2.3**	**6.3**	6.6

Table 4: *Performance of different classifiers trained on the `wav2vec2/xls-r-2b` representation. All classifiers are trained on the same data: ASVspoof'19 `train` and `dev` splits.*

4.5. What is the impact of training data?

A benefit of using a simple linear layer classifier is that we can expect the model to be robust to the number of training samples. We test this hypothesis by varying the number of training samples from ASVspoof'19 in factors of two, {2k, 4k, 8k, 16k, 32k, 50k}. In Figure 3 we show that the plots exhibit a kink at around 8k indicating that even with significantly fewer samples than all 50k, we can attain good performance on out-domain data.
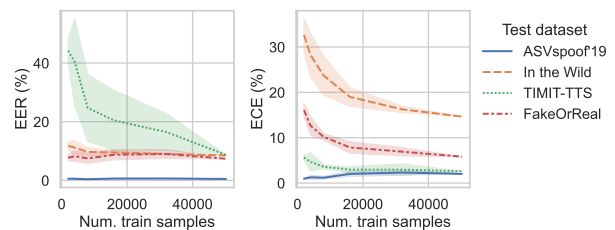

Figure 3: *Performance versus number of ASVspoof'19 training samples using `wav2vec2/xls-r-2b`. Error bars are one standard deviation over three random subsets of training data.*

5. Conclusions

We investigated self-supervised representations for audio deepfake detection in terms of two important properties: generalisation and calibration. The employed approach—self-supervised representations, followed by a linear layer—attained state-of-the-art results on a benchmark of eight out-of-domain datasets, as well as on the task of reliability estimation. In terms of the type of representations, we have showed the importance of features from large models (the 2B-parameter variant of wav2vec 2.0). In terms of the classifier, a linear layer ensured strong out-of-domain performance provided it was not highly regularised.

Limitations. Despite the strong results, our study leaves an important question open: Why do we sometimes see inconsistent results across datasets? This lack of understanding is not specific to our work, but underpins deepfake detection methods in general [37, 38]. Therefore, a worthwhile future endeavour is to provide a better understanding of what deepfake classifiers learn.

6. Acknowledgements

7. References

[1] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "AudioLDM: Text-to-audio generation with latent diffusion models," in *Proc. ICML*, 2023.

[2] M. Masood, M. Nawaz, K. M. Malik, A. Javed, A. Irtaza, and H. Malik, "Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward," *Applied Intelligence*, vol. 53, no. 4, 2023.

[3] H. Kim, S. Kim, J. Yeom, and S. Yoon, "UnitSpeech: Speaker-adaptive speech synthesis with untranscribed data," in *Proc. Interspeech*, 2023.

[4] X. Wang and J. Yamagishi, "Investigating self-supervised front ends for speech spoofing countermeasures," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2022.

[5] X. H. Zhang, J. Yi, J. Tao, C. Wang, and C. Y. Zhang, "Do you remember? Overcoming catastrophic forgetting for fake audio detection," in *Proc. ICML*, 2023.

[6] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-End anti-spoofing with RawNet2," in *Proc. ICASSP*, 2021.

[7] N. M. Müller, P. Czempin, F. Dieckmann, A. Froghyar, and K. Böttinger, "Does audio deepfake detection generalize?" in *Proc. Interspeech*, 2022.

[8] X. Wang *et al.*, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, 2020.

[9] N. M. Müller, F. Dieckmann, P. Czempin, R. Canals, K. Böttinger, and J. Williams, "Speech is silver, silence is golden: What do ASVspoof-trained models really learn?" in *Proc. ASVspoof 2021 Workshop*, 2021.

[10] S. Borzì, O. Giudice, F. Stanco, and D. Allegra, "Is synthetic voice detection research going into the right direction?" in *Proc. CVPR*, 2022.

[11] U. Ojha, Y. Li, and Y. J. Lee, "Towards universal fake image detectors that generalize across generative models," in *Proc. CVPR*, 2023.

[12] H. Tak, M. Todisco, X. Wang, J. Jung, J. Yamagishi, and N. W. D. Evans, "Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2022.

[13] Y. Xie, H. Cheng, Y. Wang, and L. Ye, "Learning a self-supervised domain-invariant feature representation for generalized audio deepfake detection," in *Proc. Interspeech*, 2023.

[14] F. Guillaro, D. Cozzolino, A. Sud, N. Dufour, and L. Verdoliva, "TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization," in *Proc. CVPR*, 2023.

[15] D. Salvi, P. Bestagini, and S. Tubaro, "Reliability estimation for synthetic speech detection," in *Proc. ICASSP*, 2023.

[16] D. Salvi, B. Hosler, P. Bestagini, M. C. Stamm, and S. Tubaro, "TIMIT-TTS: A text-to-speech dataset for multimodal synthetic media detection," *IEEE Access*, vol. 11, 2023.

[17] R. Reimao and V. Tzerpos, "For: A dataset for synthetic speech detection," in *Proc. International Conference on Speech Technology and Human-Computer Dialogue*, 2019.

[18] L. Zhang, X. Wang, E. Cooper, N. Evans, and J. Yamagishi, "The partialspoof database and countermeasures for the detection of short fake speech segments embedded in an utterance," *IEEE/ACM TASLP*, vol. 31, 2023.

[19] A. Yaroshchuk, C. Papastergiopoulos, L. Cuccovillo, P. Aichroth, K. Votis, and D. Tzovaras, "An open dataset of synthetic speech," in *Proc. Workshop on Information Forensics and Security*, 2023.

[20] N. M. Muller, P. Kawa, W. H. Choong, E. Casanova, E. Golge, T. Muller, P. Syga, P. Sperl, and K. Bottinger, "MLAAD: The multi-language audio anti-spoofing dataset," *ArXiv*, vol. abs/2401.09512, 2024.

[21] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, 2020.

[22] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T. hsien Huang, W.-C. Tseng, K. tik Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. rahman Mohamed, and H. yi Lee, "SUPERB: Speech processing universal performance benchmark," in *Proc. Interspeech*, 2021.

[23] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *ArXiv*, vol. abs/1807.03748, 2018.

[24] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," in *Proc. Interspeech*, 2021.

[25] S. Chen *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, 2022.

[26] J. Błasiok, P. Gopalan, L. Hu, and P. Nakkiran, "When does optimizing a proper loss yield calibration?" in *Proc. NeurIPS*, 2023.

[27] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," in *Advances in Large Margin Classifiers*, 2000.

[28] A. Carrell, N. Mallinar, J. Lucas, and P. Nakkiran, "The calibration generalization gap," *arXiv preprint arXiv:2210.01964*, 2022.

[29] U. Bhatt *et al.*, "Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty," in *Proc. ACM Conference on AI, Ethics, and Society*, 2021.

[30] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *Proc. ICASSP*, 2021.

[31] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proc. AAAI*, 2015.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, 2011.

[33] A. Nautsch, X. Wang, N. Evans, T. H. Kinnunen, V. Vestman, M. Todisco, H. Delgado, M. Sahidullah, J. Yamagishi, and K. A. Lee, "ASVspoof 2019: Spoofing countermeasures for the detection of synthesized, converted and replayed speech," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 3, no. 2, 2021.

[34] J. Lu, Y. Zhang, W. Wang, Z. Shang, and P. Zhang, "One-class knowledge distillation for spoofing speech detection," in *Proc. ICASSP*, 2024.

[35] Z. Zhang, X. Yi, and X. Zhao, "Fake speech detection using residual network with transformer encoder," in *Proc. ACM Workshop on Information Hiding and Multimedia Security*, 2021.

[36] M. S. A. Nadeem, J.-D. Zucker, and B. Hanczar, "Accuracy-rejection curves (ARCs) for comparing classification methods with a reject option," in *Machine Learning in Systems Biology*, 2009.

[37] L. Cuccovillo, C. Papastergiopoulos, A. Vafeiadis, A. Yaroshchuk, P. Aichroth, K. Votis, and D. Tzovaras, "Open challenges in synthetic speech detection," in *Proc. International Workshop on Information Forensics and Security*, 2022.

[38] J. Yi, C. Wang, J. Tao, X. Zhang, C. Y. Zhang, and Y. Zhao, "Audio deepfake detection: A survey," *ArXiv*, vol. abs/2308.14970, 2023.