PROBABILISTIC VERIFICATION OF VOICE ANTI-SPOOFING MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in generative models have amplified the risk of malicious misuse of speech synthesis technologies, enabling adversaries to impersonate target speakers and access sensitive resources. Although speech deepfake detection has progressed rapidly, most existing countermeasures lack formal robustness guarantees or fail to generalize to unseen generation techniques. We propose PV-VASM, a probabilistic framework for verifying the robustness of voice anti-spoofing models (VASMs). PV-VASM estimates the probability of misclassification under text-to-speech (TTS), voice cloning (VC), and parametric signal transformations. The approach is model-agnostic, builds on known concentration inequalities, and enables robustness verification against speech synthesis techniques and input perturbations. We improve existing high-confidence estimation methods with an adaptive selection mechanism without sacrificing their performance, and validate the framework across diverse experimental settings, demonstrating its effectiveness as a practical robustness verification tool.

1 INTRODUCTION

Recent advances in text-to-speech (TTS) and voice cloning (VC) have substantially improved synthetic speech quality and accessibility (Casanova et al., 2024; Azzuni & Saddik, 2025). While these technologies have many legitimate applications, realistic synthetic speech also creates security risks, including speaker impersonation and unauthorized access (Wang et al., 2025). This has motivated rapid progress in voice anti-spoofing (VAS) and speech deepfake detection.

Modern VAS systems achieve strong empirical performance through improved architectures, representations, and training procedures (Li et al., 2025a). However, their robustness remains limited under distribution shift. Evaluations such as ASVspoof (Wang et al., 2024), In-the-Wild (Müller et al., 2022), and WildSpoof (Wu et al., 2026) show substantial performance degradation on unseen spoofing methods, audio conditions, and domains. Thus, empirical accuracy alone provides limited evidence of reliability under deployment conditions. A key limitation is the lack of formal robustness guarantees (Li et al., 2023). Existing certification methods are typically designed for restricted perturbation classes and do not directly cover complex transformations induced by TTS, VC, and other speech generators. Consequently, robustness verification of VAS models against such generative transformations remains largely unexplored.

We address this problem with PV-VASM, a model-agnostic probabilistic framework for black-box robustness verification of VAS systems. PV-VASM estimates an upper bound on the probability of misclassification under an arbitrary user-defined attack distribution. It requires only the ability to sample transformed or synthesized audio and therefore supports conventional signal transformations as well as TTS and VC generators, including those unseen during training. Our approach builds on probabilistic concentration bounds and develops practical procedures for estimating the required statistics and selecting verification hyperparameters. Within the framework, we deliberately adopt a set of complementary verification bounds and adaptively select among them. Such a design is motivated by the fact that the quantity being bounded, the error probability, is unknown and strongly instance-dependent: it may vary across input audio, the set of parametric perturbations, or their parameters. We found that no single probabilistic bound dominates uniformly: specifically, for a benign label-preserving transformation a bound can be tight, but become excessively conservative for aggressive transformations or generative models, or vice versa. Consequently, we improve existing

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

estimation procedures with the adaptive AdaVAU method. Experiments across diverse transformations and speech generators demonstrate that PV-VASM provides informative error-probability bounds and complements standard empirical evaluation.

The contributions of this work are summarized as follows:

- We formalize probabilistic robustness verification for VAS and parametric transformations and generative models, allowing user-defined attack spaces.
- We study the existing concentration inequalities and derive practical procedures for estimating their statistics and hyperparameters.
- We improve existing high-confidence estimation methods with the adaptive selection mechanism, AdaVAU, without sacrificing the performance of the source anti-spoofing model or confidence guarantees of the underlying methods.
- We evaluate PV-VASM across diverse transformations and speech generators, demonstrating its applicability to pre-deployment robustness assessment of VAS systems. Our code is anonymously available at `https://anonymous.4open.science/r/PV-VASM/`.

2 Related Work

Common voice anti-spoofing (VAS) architectures include graph-based neural approaches, such as AASIST (Jung et al., 2022), which apply graph attention mechanisms to temporal and spectral representations derived, for example, using Sinc convolutions (Ravanelli & Bengio, 2018). Substantial performance gains have also been achieved by leveraging self-supervised learning (SSL) audio encoders, including Wav2Vec 2.0 (Baevski et al., 2020) and WavLM (Chen et al., 2022), as front-end feature extractors. These representations are subsequently processed by back-end architectures ranging from AASIST (e.g., Wav2Vec2–AASIST (Tak et al., 2022)) to simpler pooling-based models with linear classifiers (Aliyev & Kondratev, 2024). Further improvements have been reported through dataset diversification, model ensembling, one-class learning, specialized architectures and loss functions, and audio large language models (Foret et al., 2020; Ding et al., 2023; Borodin et al., 2024; Gu et al., 2025; Dvirniak et al., 2026).

Despite these advances, VAS systems frequently exhibit limited robustness and generalization beyond the training distribution. This behavior has been observed in major evaluation campaigns, including the ASVspoof challenges (Todisco et al., 2019; Wang et al., 2020; Yamagishi et al., 2021; Wang et al., 2024) and the ADD series (Yi et al., 2022; 2023), where performance often degrades when evaluation data contain previously unseen perturbations or speech generation methods. VAS models are also vulnerable to additive noise, signal-level transformations, and adversarial perturbations (Rabhi et al., 2024; Li et al., 2025b; Szegedy, 2013; Goodfellow et al., 2014). More generally, such degradation under distribution shift is well documented for neural networks (Singhal et al., 2023; Sehwag et al., 2019; Croce et al., 2020; Hendrycks & Dietterich, 2019).

Robustness defenses can be broadly divided into empirical and certified approaches. Empirical defenses, such as adversarial training (Wong et al., 2020; Madry, 2017; Fort & Lakshminarayanan, 2024), can provide strong empirical robustness but do not guarantee model behavior against unseen or adaptive perturbations. Certified robustness methods (Lecuyer et al., 2019; Li et al., 2023), in contrast, provide provable deterministic or probabilistic guarantees under a predefined threat model, typically specified by a transformation family and its parameters. However, limited attention has been devoted to certification of VAS systems, and existing techniques are generally not applicable to non-analytical transformations such as synthetic speech generation.

Probabilistic verification provides an alternative by estimating whether a specified robustness property holds under sampled transformations. Existing methods (Mohapatra et al., 2020; Tit et al., 2021; Baluta et al., 2021; Pautov et al., 2022) estimate the probability of model failure under a prescribed transformation distribution. In particular, CC-Cert (Pautov et al., 2022) derives an upper bound on misclassification probability using an empirical form of the Chernoff-Cramer concentration inequality (Boucheron et al., 2003). However, these approaches do not directly provide guarantees for samples produced by artificial or neural speech generators and may rely on distribution statistics, such as the coefficient of variation, whose invariance across transformations is generally not guaranteed and can lead to overly conservative estimates.

3 METHODOLOGY

In this section, we formulate the problem in mathematical terms. The following sections provide the formal description of three methods used as a basis, as well as the description of the adaptive method AdaVAU, which alternates among the aforementioned methods. We recall that AdaVAU is designed to estimate the probability of the voice anti-spoofing model to misclassify the transformed input audio. In the subsequent sections, we generalize problem formulation to the verification of model's robustness against the speech generation methods. Here and below, we treat the VAS problem as a binary classification task.

3.1 PROBLEM SETUP

Let $\mathcal{X} \subset \mathbb{R}^d$ be the space of input audios, $x \in \mathcal{X}$ be the fixed input audio, $f : \mathbb{R}^d \to \mathbb{R}^2$ be the source voice anti-spoofing model that assigns the vector of model confidences to x in the form

$$f(x) = (p_1, p_2)^\top, \tag{1}$$

where $p_1 + p_2 = 1,\ p_1, p_2 \geq 0$. Here, p_1 represents the confidence of the model to assign x to the class of spoof audio, and p_2 represents the confidence of x being classified as a bona fide audio. To ease the notation, we introduce the classification rule in the form

$$h(x) = \arg\max_i f_i(x). \tag{2}$$

In this work, we focus on the problem of verifying the robustness of f under parametric perturbations of its input. Formally, the parametric transformation $\phi : \mathcal{X} \times \Theta \to \mathcal{X}$ is the mapping of input audio space to itself, where Θ is a fixed space of parameters of the transformation ϕ. Under transformation ϕ, the input object x becomes the random variable, since it is a function of θ:

$$x' = \phi(x, \theta), \quad \theta \overset{\mu}{\sim} \Theta, \tag{3}$$

where μ is the measure on Θ, which is fixed by the design (any μ may be chosen and fixed in advance; its choice would be discussed later). The prediction of the source model on the perturbed audio becomes the random variable in the form

$$f(x') = (p_1', p_2')^\top. \tag{4}$$

The robustness of f under label-preserving transformation ϕ at point x is reflected by the probability to assign the same class index to x and x' in the form

$$\mathbb{P}_{\theta \sim \Theta}\left[h(x) = h(x')\right], \text{ where } x' = \phi(x, \theta). \tag{5}$$

Here and in what follows we omit μ for brevity. In the case of a binary classification task,

$$h(x) = h(x') \leftrightarrow\ (p_1 - 1/2)\,(p_1' - 1/2) > 0 \leftrightarrow (p_2 - 1/2)\,(p_2' - 1/2) > 0.$$

Without loss of generality, we assume that the initial audio x is correctly classified by f as a bona fide one, so $p_2 > 1/2$. Thus, $h(x) = h(x') \leftrightarrow p_2' > 1/2$, and

$$\mathbb{P}_{\theta \sim \Theta}\left[h(x) \neq h(x')\right] = \mathbb{P}_{\theta \sim \Theta}[p_2' < 1/2]. \tag{6}$$

In case of a nontrivial μ and f, the probability from Eq. 6 is intractable. The goal is to design a method that will provide tight upper bound for this probability and, consequently, a tight upper bound for the probability of f to misclassify x'.

3.2 BASIS BOUNDS

In this section, we introduce the basis verification procedures that are subsequently integrated into AdaVAU, the approach we propose. The choice of the basis procedures is made to utilize complementary verification principles. Specifically, we both (i) apply exponential concentration inequalities to continuous confidence scores that can be very tight when the mislcasification probability is small and (ii) estimate error indicator, directly penalizing empirical variance, which can be meaningful when the score distribution is heavy-tailed.

Here and below, we assume that the input x is fixed. To ease the notation, we introduce the random variable $Z \equiv p_2'$ and $p_{\text{miss}} \equiv \mathbb{P}_{\theta \sim \Theta}\left[h(x) \neq h(x')\right]$ which is a function of x.

3.2.1 CC-CERT

Following (Pautov et al., 2022), we apply Markov's inequality to exponential transformed random variable e^{tZ} for $t < 0$ and obtain

$$p_{\text{miss}} \leq \inf_{t<0} \mathbb{E}(e^{tZ})e^{-t/2}, \tag{7}$$

where $t \in \mathbb{R}$. Since the expectation from Eq. 7 is intractable it can be upper-bounded it via sampling the random variable Z and computing the statistics of the resulting sample. Specifically, for the given $t < 0$, let

$$Y_j = \frac{1}{n}\sum_{i=1}^{n} \exp(tZ_i^j)\exp(-t/2),\ j \in [1, \ldots, k] \tag{8}$$

be the set of k independent and identically distributed sample means, each computed over n realizations of Z^j (here, superscript j denotes the batch of realizations of the random variable used to compute Y_j). Then, for all $d \in (0, 1)$, the statistic

$$\mathcal{A}(x) = \max\{Y_1, \ldots, Y_k\}\, d^{-1} \tag{9}$$

is an upper bound for the expectation $\mathbb{E}(e^{tZ})e^{-t/2}$ with high probability. Specifically, the error probability of the method is upper-bounded by

$$\mathbb{P}_{\theta\sim\Theta}\left[\mathcal{A}(x) < \mathbb{E}(e^{tZ})e^{-t/2}\right] \leq \left(\frac{1}{1 + n(1-d)^2c^{-2}}\right)^k \equiv p(n, k, c), \tag{10}$$

where

$$c = \frac{\sqrt{\mathbb{V}(e^{tZ})}}{\mathbb{E}(e^{tZ})} \tag{11}$$

is the coefficient of variation of the random variable e^{tZ}. Informally, given $m = n \times k$ realizations of a random variable Z, the statistic $\mathcal{A}(x)$ provides an upper-bound for the probability of misclassifying x', that holds with high probability.

In the following section, we describe the procedure to estimate the coefficient of variation from Eq. 11 and the procedure to dynamically choose the value of hyperparameter d from Eq. 9, treating it as a function of desired confidence level.

3.2.2 ADAPTIVE PARAMETER ESTIMATION

In Eq. 9, a high-probability upper bound for the value from Eq. 6 is computed. Here and below, we say that the method A *makes an error* given x if its output underestimates an unknown error probability, namely, if

$$\mathcal{A}(x) < p_{\text{miss}}. \tag{12}$$

To estimate the probability of error from Eq. 12, given the sample $\{\theta_j\}_{j=1}^m \sim \Theta$, one has first to estimate the right-hand side of Eq. 10. Since both the mean and variance of the random variable e^{tZ} are intractable, one can estimate the coefficient of variation using $m = n \times k$ realizations of the random variable Z. In our approach, we use one-sided confidence interval estimation of the coefficient of variation in the form of the modified McKay's approximation (Payton, 1996):

$$\mathbb{P}_{\theta\sim\Theta}\left[c > \left(\frac{\chi^2_{\alpha/4}(1+\hat{c}^2)}{m\hat{c}^2}\right)^{-1/2}\right] < \delta/2. \tag{13}$$

Here $\delta \in (0, 1)$ is the aforementioned desired confidence level, $\chi^2_{\alpha/4}$ is the lower $\alpha/4-$percentile of the Chi-square distribution with $m - 1$ degrees of freedom, and $\hat{c}$ is the sample coefficient of variation.

Note that by combining Eq. 10 and Eq. 13, one can estimate the probability of an error of the algorithm. Specifically, by introducing an auxiliary random variable

$$A = \mathbb{1}\left\{p_{\text{miss}} > \mathcal{A}(x)\right\}, \tag{14}$$

where $\mathbb{1}\{\cdot\}$ is an indicator function, and setting $\tilde{c} = \left(\chi^2_{\alpha/4}(1+\hat{c}^2)m^{-1}\hat{c}^{-2}\right)^{-1/2}$ from Eq. 13, one can upper bound the probability from Eq. 6 by

$$\mathbb{P}[A=1] < 1 \times \mathbb{P}[c > \tilde{c}] + p(n,k,c=\tilde{c})\mathbb{P}[c \leq \tilde{c}] < \delta/2 + p(n,k,c=\tilde{c}). \tag{15}$$

To bound misclassification probability from Eq. 6 with the desired confidence level δ, we choose the value of d so that $p(n,k,c=\tilde{c}) \leq \delta/2$. Solving this inequality, we obtain the condition on d :

$$d \leq 1 - \widetilde{c}\sqrt{\frac{(2/\delta)^{1/k}-1}{n}} := d_{\text{opt}}. \tag{16}$$

It is worth noting that for sufficiently large values of c, McKay's approximation in the form from Eq. 13 should be replaced by, for example, bootstrap-based interval estimation. Recall that the definition of $\mathbb{P}[A=1]$ from Eq. 14 is given for the verification of an initially bona fide audio. To verify robustness to the transformations of initially spoof audio, we flip the sign of parameter t from Eq. 7: $t \mapsto -t$.

3.2.3 Empirical Bernstein's inequality

Misclassification probability can be reformulated as an expectation of indicator of the event

$$p_{\text{miss}} = \mathbb{E}_{\theta\sim\Theta}E(\theta), \tag{17}$$

where $E(\theta) = \mathbb{1}\{h(x) \neq h(\phi(x,\theta))\}$. Note that $E(\theta)$ is a random variable drawn from a Bernoulli distribution with the unknown parameter p_{miss}. Since $0 \leq E(\theta) \leq 1$, one can use Empirical Bernstein's inequality (Maurer & Pontil, 2009) to bound the probability of deviation of its point-wise estimate from its unknown true value. Specifically, for budget $m > 2$, we define following variables

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{m} E_i \qquad s_E^2 = \frac{m}{m-1}\widehat{p}(1-\widehat{p}), \qquad \rho_{\text{EB}} = \sqrt{\frac{2s_E^2 L}{m}} + \frac{7L}{3(m-1)}, \tag{18}$$

where $L = \ln(2/\delta)$, and $\widehat{p}$ is a sample average; we call it an empirical misclassification probability. Then, according to Empirical Bernstein's inequality, we have

$$p_{\text{miss}} \leq \widehat{p} + \rho_{\text{EB}} \equiv \mathcal{A}_{\text{EB}} \tag{19}$$

with probability at least $1-\delta$.

3.2.4 Bernoulli–Cantelli inequality

Another way to bound $\mathbb{E}_{\theta\sim\Theta}E(\theta)$ is via the Cantelli inequality (Cantelli, 1929) for a Bernoulli random variable $E(\theta)$. Given the budget $m = nk$, we define

$$Y_j^E = \frac{1}{n}\sum_{i=(j-1)n+1}^{jn} E_i, \qquad q_E = \max_{1\leq j\leq k} Y_j^E, \qquad a = \frac{\delta^{-1/k}-1}{n}. \tag{20}$$

Then, for $u < p$, from Cantelli's inequality, using the independence of Y_j^E, one can bound

$$\mathbb{P}\left\{\max_j Y_j^E \leq u\right\} \leq \left(\frac{p(1-p)}{p(1-p)+n(p-u)^2}\right)^k, \quad \text{where } p = p_{\text{miss}}(x). \tag{21}$$

Setting

$$\mathcal{A}_{\text{BC}}(x) := \frac{2q_E + a + \sqrt{a^2 + 4aq_E(1-q_E)}}{2(1+a)} \tag{22}$$

yields an estimation that upper bounds an unknown value p_{miss} with the desired confidence:

$$\mathbb{P}\{p_{\text{miss}}(x) \leq \mathcal{A}_{\text{BC}}(x)\} \geq 1-\delta. \tag{23}$$

3.3 AdaVAU

In this section, we describe the adaptive selection procedure, AdaVAU (or AVU). Specifically, having the whole budget $\mathcal{M} = \{\theta_j\}_{j=1}^m$ of parameters, we randomly subsample a small set $\mathcal{P} \subset \mathcal{M}$ to and estimate the value $\hat{p}$ from Eq. 18 over $\mathcal{P}$:

$$\mathcal{A}_{\text{AV}}(\mathcal{M}) = \begin{cases} \mathcal{A}_{\text{CC}}(\mathcal{M} \setminus \mathcal{P}), & \text{if } \widehat{p}(\mathcal{P}) < \varepsilon_{\text{TH1}} \text{ and } \nVdash\{min_{\theta \in \mathcal{P}} |\theta - 1/2| > g\} \\ \mathcal{A}_{\text{BC}}(\mathcal{M} \setminus \mathcal{P}), & \text{if } \varepsilon_{\text{TH1}} \leq \widehat{p}(\mathcal{P}) < \varepsilon_{\text{TH2}}, \\ \mathcal{A}_{\text{EB}}(\mathcal{M} \setminus \mathcal{P}), & \text{otherwise}, \end{cases} \tag{24}$$

where thresholds $\varepsilon_{\text{TH1}}, \varepsilon_{\text{TH2}}$ and gap g are set heuristically. The verification procedure against parametric transformations is described in Appendix B.3.

3.4 Adaptation to Generative Models

In the preceding sections, we considered robustness verification against parametric transformations that preserve the semantic class of the original audio signal. In this subsection, we extend the proposed framework to enable robustness verification of voice anti-spoofing models against generative models, specifically text-to-speech (TTS) and voice-cloning (VC) systems.

In the TTS setting, the objective is no longer the robustness verification with respect to transformations of a fixed input audio sample x, but rather to reason about an entire family (distribution) of audio signals generated by a TTS model. The generated audio depends primarily on the input text and, potentially, on additional generation parameters such as speech rate or selected artificial voice:

$$\phi(x, \theta) \mapsto g(t, \theta), \tag{25}$$

where g denotes the considered TTS model, $t \sim \mathcal{T}$, with $\mathcal{T} \subset \mathbb{R}^{l \times s}$ denoting a text representation of vocabulary size s and length l, and $\theta \sim \Theta$ represents the collection of additional independent generation variables. Such variables may be multidimensional and either be fixed or drawn from the corresponding distributions; they can include, for example, speech rate, voice identity, or language configuration of the TTS system, in the case of multi-speaker or multilingual TTS. In practice, the text inputs are often drawn from natural language corpora or task-specific datasets.

Unlike sample-specific verification against transformations of the given input x, the key idea in this setting is to conduct robustness verification with respect to the *distribution of audio signals induced by the generative model* g. That is, we aim to quantify how frequently the classifier f assigns a generated sample $x' = g(\theta)$ to the incorrect class of bona fide speech: $\mathbb{P}[p'_2 > 1/2]$.

PV-VASM can be used to provide verification results for voice cloning (VC) systems. Voice cloning aims to synthesize speech for an arbitrary text t while preserving the voice characteristics of a reference speaker. In a nutshell, a voice cloning model g_{vc} can be expressed as

$$g_{\text{vc}} : \mathcal{T} \times \mathcal{X} \times \Theta \to \mathcal{X}, \quad x' = g_{\text{vc}}(t, x_{\text{ref}}, \theta), \tag{26}$$

where $t \in \mathcal{T}$ denotes the input text, $x_{\text{ref}} \in \mathcal{X}$ is a reference audio sample of the target speaker s_{tgt}, and $\theta \in \Theta$ represents additional generation parameters. This formulation naturally supports both sample-specific and distribution-level robustness verification. In the sample-specific setting, for a fixed input x (which need not necessarily be bona fide speech), one can sample or segment the reference audio x_{ref}, as well as vary the input texts and additional generation parameters. In contrast, distribution-level verification further considers variability over the entire set of admissible input audio signals x, thereby assessing robustness with respect to the full data-generating distribution. Note that VC is a label-switching transformation for the bona fide inputs.

4 Experimental setup

Source Model We selected Wav2Vec2-AASIST as the architecture for the source model f. This model was trained for two epochs using cross-entropy loss and the AdamW (Loshchilov & Hutter, 2017) optimizer, the best checkpoint was selected according to validation results. As the experiments are computationally expensive, we restrict the evaluation to a single model. This model remains strongly competitive with existing methods and is sufficient to demonstrate the feasibility of PV-VASM, which can be applied to verify arbitrary models and which does not aim at improving the classification metrics.

Table 1: Robustness verification results against parametric input transformations at nominal failure probability $\alpha = 10^{-6}$ and budget $m = 20000$. All reported metrics are percentages, evaluated on the 298 initially correctly classified audios out of 300; the PCA thresholds ε are probabilities. CCC: CC-Cert; EBI: Empirical Bernstein's inequality; BCI: Bernoulli–Cantelli inequality; AVU: Adaptive Voice Antispoofing error upperbounding.

Transform	Method	Mean (%)				PCA (%)			
		$\mathcal{A}(x)$	$r(x)$	$\widehat{p}(x)$	Cost	$\varepsilon = 10^{-5}$	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-2}$	$\varepsilon = 0.05$
LPF	CCC	0.133	0.093	0.040	0.088	98.322	98.993	99.664	99.664
	EBI	0.213	0.173	0.040	0.168	0.000	0.000	99.664	99.664
	BCI	0.194	0.155	0.040	0.149	0.000	0.000	99.664	99.664
	AVU	0.046	0.006	0.040	0.001	98.322	98.993	99.664	99.664
HPF	CCC	10.114	2.765	7.349	2.622	85.570	87.919	88.591	88.926
	EBI	7.640	0.291	7.350	0.149	0.000	0.000	88.926	88.926
	BCI	7.734	0.384	7.350	0.242	0.000	0.000	88.926	88.926
	AVU	7.501	0.152	7.349	0.009	85.570	87.248	88.926	88.926
Pitch Shift	CCC	33.789	12.314	21.475	11.755	59.060	61.745	62.081	62.752
	EBI	22.138	0.668	21.470	0.104	0.000	0.000	63.423	63.423
	BCI	22.551	1.081	21.470	0.517	0.000	0.000	63.423	63.423
	AVU	22.075	0.598	21.477	0.041	59.060	61.745	63.423	63.423
BPF	CCC	20.964	12.733	8.231	12.202	50.336	52.349	55.369	61.409
	EBI	8.851	0.621	8.229	0.089	0.000	0.000	60.067	72.483
	BCI	9.214	0.985	8.229	0.452	0.000	0.000	59.396	71.141
	AVU	8.801	0.571	8.230	0.039	49.664	50.336	60.067	72.483
Gain	CCC	0.094	0.063	0.031	0.056	99.329	99.329	99.329	99.329
	EBI	0.206	0.175	0.031	0.168	0.000	0.000	99.329	99.664
	BCI	0.193	0.161	0.031	0.155	0.000	0.000	99.329	99.329
	AVU	0.038	0.007	0.031	0.000	99.329	99.329	99.329	99.664
Composite	CCC	12.893	8.936	3.957	8.665	66.443	67.785	70.470	75.168
	EBI	4.343	0.384	3.959	0.115	0.000	0.000	82.886	87.919
	BCI	6.098	2.139	3.959	1.870	0.000	0.000	79.530	83.893
	AVU	6.202	2.244	3.958	1.974	65.436	66.779	79.195	84.899

Datasets and augmentations. For the train data, a combination of open-source datasets was used, including ASVspoof 19, 21 (LA and DF) splits, ASVspoof 5, ADD $22-23$, DFADD (Du et al., 2024), SONAR (Li et al., 2024), CFAD (Ma et al., 2024), MLAAD (Müller et al., 2024), Speech-to-Latex (Korzh et al., 2025), and Mozilla Common Voice (Ardila et al., 2019). During training, to improve the model's empirical robustness, each audio sample was subjected to a composition of randomly selected augmentations applied in a random order. Augmentations were chosen from the following set: voice activity detection, random signal cropping, background noise addition, room impulse response (RIR) simulation, Gaussian noise injection, band-, low-, and high-pass filter (BPF, LPF, HPF), loudness normalization, codec augmentation, time stretching, pitch shifting, bit crushing, and gain adjustment. For the test data, a balanced class subset of 300 audio samples from the ASVspoof 5 evaluation (test) subset was used. ASVspoof5 serves as a modern benchmark to demonstrate feasibility. Only initially correctly classified audio samples were used for the verification, following the practice in the literature.

Default hyperparameters. The following default values of hyperparameters were used in the experiments unless said otherwise: the value of δ from Eq. 9 was set to 0.9, the range of parameter t from Eq. 7 was set to be $[-50, -10^{-4}]$. The number of transformations of input samples n to compute the single statistic from Eq. 8, the total number of statistics k, and an upper bound α for the error probability of the method from Eq. 15 are varied for different experiments and are discussed in the subsequent sections.

Conventional transformations. We evaluate PV-VASM in two different settings, namely, against conventional input perturbations and against speech generation models. In the first setting, we considered parametric perturbations *that do not change the predicted class of the input object* x. For this purpose, several input transformations that are commonly applied as audio augmentation were used. In a nutshell, if the input audio is a deepfake, the method should verify that the classifier's prediction remains the same under an input transformation and analogously for the bona fide audio. Details on the used conventional audio transformations are presented in Appendix B.1. In our experiments, the parameters of input transformations were sampled uniformly from the selected parameter's range or among available noise and RIR samples; however, all the parameters may be sampled from any other analytical distribution. We indicate that the ranges of input transformations during the verification procedure differ from the ones used during training, and are presented in the subsequent section.

Speech generators. To evaluate the effectiveness of the method against TTS and VC methods, we generated data using the following speech generation systems: Vosk TTS (Alpha Cephei Inc, 2026), Silero TTS (Silero Team, 2025; daswer123, 2026), Coqui XTTS-v2 (Coqui, 2026; Casanova et al., 2024), F5-TTS (Chen et al., 2025), CosyVoice (CosyVoice), and the proprietary services ElevenLabs (ElevenLabs, 2026) and FineVoice (FineVoice, 2026). Texts used for the artificial annotation were primarily selected from the Mozilla Common Voice dataset.

Metrics. The primary quantitative measure of efficiency of a probabilistic verification method $\mathcal{A}$ in a classification problem is probabilistically certified accuracy (Pautov et al., 2022), PCA, defined as

$$\mathrm{PCA}(\varepsilon, \alpha, \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(x,y)\in\mathcal{D}} \mathbb{1}\{h(x) = y \wedge \mathcal{A}(x) < \varepsilon \wedge p < \alpha/2\}, \tag{27}$$

where p is from Algorithm 1. Informally, PCA is the fraction of objects from $\mathcal{D}$ which (i) are correctly classified by the source model f; (ii) are assigned the misclassification probability $\mathcal{A}(x)$ less than the threshold value ε; (iii) have the $p(x)$ is upper bounded by $\alpha/2$, given that $\tilde{c}$ is estimated with $1 - \alpha/2$ confidence. When verification is performed with respect to the TTS-induced distribution, the resulting PCA metric is binary valued and is defined as:

$$\mathrm{PCA}(\varepsilon, \alpha) = \mathbb{1}\{\mathcal{A} \prec \varepsilon \wedge\ p \prec \alpha/2\}, \tag{28}$$

where $\mathcal{A} \prec \varepsilon$ means that $\mathcal{A}$ is uniformly less than ε.

5 Results

Parametric Input Transformations. In Table 1, we present the subset of results (the full version Table 4 and transformations' parameters Table 5 are in Appendix) of robustness verification of the considered Wav2Vec2-AASIST model against parametric input transformations. For each transformation and corresponding space of parameters, we report the number of input transformations n to compute the single statistic from Eq. 8 and the total number of statistics k; for the given value of threshold ε, we present the values of PCA metric; additionally, we report the average value of $\overline{\mathcal{A}(x)}$ from Algorithm 1 and the average value $\overline{p} = \overline{p}(n, k, \tilde{c}(x))$ from Eq. 10.

TTS. Table 2 reports robustness verification against different TTS generators. The reported p values estimate the error probability from Eq. 10 for the distribution induced by each generator. We highlight that in this setting, the verification of the VAS model's robustness against the *distribution* of perturbations induced by a TTS model is performed, so the output of the algorithm and its error probability depend on the distribution. We compare the base model with the same model finetuned on 10% of samples generated by the corresponding TTS model, while the remaining 90% are used for verification. Figure 1 in the Appendix further illustrates the verification behavior for Vosk at fixed $m = 130000$ and $\varepsilon = 0.1$ over different (n, α) pairs. The results show that robustness verification against TTS generators is more challenging than against parametric input transformations, mainly due to larger values of $\mathcal{A}$. Finetuning consistently improves the verification results, reducing $\mathcal{A}$ by approximately 1.5-3 times.

Table 2: Robustness verification results against TTS generators using the CC-Cert backbone. The confidence parameter is set to $\alpha = 10^{-6}$.

TTS model	n	k	δ	Pre-finetuning $(\mathcal{A},\ p)$	Post-finetuning $(\mathcal{A},\ p)$
Vosk	5000	26	0.75	$(0.1352,\ 3.37 \times 10^{-17})$	$(0.0686,\ 8.1 \times 10^{-7})$
Silero	650	20	0.50	$(0.5030,\ 6.1 \times 10^{-21})$	$(0.0579,\ 1.64 \times 10^{-5})$
CosyVoice	500	40	0.90	$(0.5236,\ 2.4 \times 10^{-8})$	$(0.2656,\ 3.2 \times 10^{-4})$
F5-TTS	10000	20	0.90	$(0.4019,\ 2.6 \times 10^{-25})$	$(0.2319,\ 1.7 \times 10^{-16})$
ElevenLabs	500	6	0.50	$(0.3308,\ 3.0 \times 10^{-5})$	$(0.2002,\ 3.9 \times 10^{-3})$
FineVoice	200	25	0.50	$(0.1446,\ 3.8 \times 10^{-7})$	$(0.0721,\ 1.7 \times 10^{-5})$

Voice Cloning. Similarly to the TTS experiment, we measured the model's robustness against VC models. We present the results of verification of robustness against XTTSv2 and F5 models in Table 3. One can notice that the improvement of PCA for the XTTSv2 model after finetuning is slightly less than the one for the f5 model; this can be explained by the fact that some inclusion of samples generated by the XTTSv2 model is present in the training dataset of the base model f.

Table 3: Robustness verification results against VC generators using the CC-Cert backbone. Verification parameters are set to $\alpha = 10^{-4}$ and $\delta = 0.5$.

Stage	VC model	n	k	PCA ($\varepsilon = 0.05$)	PCA ($\varepsilon = 0.10$)	PCA ($\varepsilon = 0.30$)	$\overline{\mathcal{A}(x)}$	$\overline{p(x)}$
Pre-finetuning	XTTSv2	100	50	0.92	0.94	1.00	0.0148	3.8×10^{-6}
	F5-TTS	500	10	0.00	0.00	0.24	0.4167	9.6×10^{-5}
Post-finetuning	XTTSv2	100	50	0.96	0.96	1.00	0.0078	4.1×10^{-6}
	F5-TTS	500	10	0.00	0.00	0.30	0.3531	6.8×10^{-4}

6 Discussion and Limitations

PV-VASM allows users to define an arbitrary attack distribution, provided that i.i.d. samples can be drawn from it. This enables verification against transformations, generators, codecs, and their combinations without modifying the evaluated model. Verification is strongest for relatively mild transformations, such as LPF with a high cutoff frequency or moderate gain, and degrades for more severe perturbations, including strong background noise, restrictive BPF settings, and composite attacks. For TTS and VC generators, the results further reveal limited out-of-domain robustness of VAS models, while finetuning on the corresponding domains consistently improves verification. High variance of the random variable e^{tZ} in Eq. 24 leads to looser bounds and more conservative PCA values, particularly for complex generative transformations. Finally, bound tightness depends on the parameters and hyperparameters of the underlying verification method, which should therefore be selected carefully.

7 Conclusion and Future Work

In this paper, we proposed PV-VASM, a framework for robustness verification of voice anti-spoofing models. We theoretically derived an upper bound for the error probability of the method and experimentally demonstrated the verification results in different settings, including the presence of parametric input perturbations, text-to-speech generation models, and voice cloning methods. We showed that the robustness of voice anti-spoofing models crucially depends on the type of perturbation and the width of its parameter space. We therefore confirmed that robustness to a simple parametric perturbation is noticeably higher than that to perturbations produced by a speech generation model. The proposed method can be applied to verify the robustness of models before real-world applications. Future work includes optimization of the selection procedure, AdaVAU: for example, considering multiple statistics from the pilot subset may form a basis for a more optimal selection mechanism. Additionally, the values of some hyperparameters (i.e., threshold for AdaVAU) were estimated empirically, so the dependency of the methods on the hyperparameters can be extended beyond the ablation study.

AI USE STATEMENT

AI models (ChatGPT) and tools (Grammarly) were used only for text polishing and shortening.

REPRODUCIBILITY STATEMENT

The main algorithms and parameters are discussed in the text. Source code is anonymously available.

REFERENCES

Ali Aliyev and Alexander Kondratev. Intema system description for the asvspoof5 challenge: power weighted score fusion. In *Proc. ASVspoof 2024*, pp. 152–157, 2024.

Alpha Cephei Inc. Vosk TTS. PyPI, 2026. URL https://pypi.org/project/vosk-tts/. Accessed: 2026-09-22.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*, 2019.

Hussam Azzuni and Abdulmotaleb El Saddik. Voice cloning: Comprehensive survey. *arXiv preprint arXiv:2505.00579*, 2025.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

Teodora Baluta, Zheng Leong Chua, Kuldeep S Meel, and Prateek Saxena. Scalable quantitative verification for deep neural networks. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 312–323. IEEE, 2021.

Kirill Borodin, Vasiliy Kudryavtsev, Dmitrii Korzh, Alexey Efimenko, Grach Mkrtchian, Mikhail Gorodnichev, and Oleg Y. Rogov. AASIST3: KAN-enhanced AASIST speech deepfake detection using SSL features and additional regularization for the ASVspoof 2024 Challenge. In *The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)*, pp. 48–55, 2024. doi: 10.21437/ASVspoof.2024-8.

Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In *Summer school on machine learning*, pp. 208–240. Springer, 2003.

Francesco Paolo Cantelli. Sui confini della probabilita. In *Atti del Congresso Internazionale dei Matematici: Bologna del 3 al 10 de settembre di 1928*, pp. 47–60, 1929.

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. *arXiv preprint arXiv:2406.04904*, 2024.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, 2022.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6255–6271, 2025.

Coqui. XTTS-v2. Hugging Face model card, 2026. URL https://huggingface.co/coqui/XTTS-v2.

CosyVoice. CosyVoice. GitHub repository, 2026. URL https://github.com/FunAudioLLM/CosyVoice.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

daswer123. Silero TTS. PyPI, 2026. URL `https://pypi.org/project/silero-tts/`.

Siwen Ding, You Zhang, and Zhiyao Duan. Samo: Speaker attractor multi-center one-class learning for voice anti-spoofing. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Jiawei Du, I-Ming Lin, I-Hsiang Chiu, Xuanjun Chen, Haibin Wu, Wenze Ren, Yu Tsao, Hung-yi Lee, and Jyh-Shing Roger Jang. Dfadd: The diffusion and flow-matching based audio deepfake dataset. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pp. 921–928. IEEE, 2024.

Artem Dvirniak, Evgeny Kushnir, Dmitrii Tarasov, Artem Iudin, Oleg Kiriukhin, Mikhail Pautov, Dmitrii Korzh, and Oleg Y Rogov. Towards robust speech deepfake detection via human-inspired reasoning. *arXiv preprint arXiv:2603.10725*, 2026.

ElevenLabs. ElevenLabs. `https://elevenlabs.io`, 2026.

FineVoice. FineVoice: AI voice generator. Official website, 2026. URL `https://finevoice.ai/`.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Stanislav Fort and Balaji Lakshminarayanan. Ensemble everything everywhere: Multi-scale aggregation for adversarial robustness. *arXiv preprint arXiv:2408.05446*, 2024.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Hao Gu, Jiangyan Yi, Chenglong Wang, Jianhua Tao, Zheng Lian, Jiayi He, Yong Ren, Yujie Chen, and Zhengqi Wen. Allm4add: Unlocking the capabilities of audio large language models for audio deepfake detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 11736–11745, 2025.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Iver Jordal, Shahul ES, Hervé BREDIN, Kento Nishi, Francis Lata, Harry Coultas Blum, Pariente Manuel, Akash Raj, Keunwoo Choi, Piotr Żelasko, et al. Asteroid-team/torch-audiomentations: v0. 11.0. *Zenodo*, 2022.

Iver Jordal, Araik Tamazian, Tushar Dhyani, Emmanouil Theofanis Chourdakis, Nikolay Karpov, Christopher Landschoot, Céline Angonin, Omer Sarioglu, Will Drevo, Fabio Catania, et al. iver56/audiomentations: v0. 42.0. *Zenodo*, 2025.

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 *ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6367–6371. IEEE, 2022.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur. A study on data augmentation of reverberant speech for robust speech recognition. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5220–5224. IEEE, 2017.

Dmitrii Korzh, Dmitrii Tarasov, Artyom Iudin, Elvir Karimov, Matvey Skripkin, Nikita Kuzmin, Andrey Kuznetsov, Oleg Y Rogov, and Ivan Oseledets. Speech-to-latex: New models and datasets for converting spoken equations and sentences. *arXiv preprint arXiv:2508.03542*, 2025.

Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 656–672. IEEE, 2019.

Linyi Li, Tao Xie, and Bo Li. Sok: Certified robustness for deep neural networks. In *2023 IEEE symposium on security and privacy (SP)*, pp. 1289–1310. IEEE, 2023.

Menglu Li, Yasaman Ahmadiadli, and Xiao-Ping Zhang. A survey on speech deepfake detection. *ACM Computing Surveys*, 57(7):1–38, 2025a.

Xiang Li, Pin-Yu Chen, and Wenqi Wei. Sonar: A synthetic ai-audio detection framework and benchmark. *arXiv preprint arXiv:2410.04324*, 2024.

Xiang Li, Pin-Yu Chen, and Wenqi Wei. Measuring the robustness of audio deepfake detectors. *arXiv preprint arXiv:2503.17577*, 2025b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Haoxin Ma, Jiangyan Yi, Chenglong Wang, Xinrui Yan, Jianhua Tao, Tao Wang, Shiming Wang, and Ruibo Fu. Cfad: A chinese dataset for fake audio detection. *Speech Communication*, 164: 103122, 2024.

Aleksander Madry. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

Jeet Mohapatra, Ching-Yun Ko, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. Higher-order certification for randomized smoothing. *Advances in Neural Information Processing Systems*, 33:4501–4511, 2020.

Nicolas M Müller, Pavel Czempin, Franziska Dieckmann, Adam Froghyar, and Konstantin Böttinger. Does audio deepfake detection generalize? *Interspeech*, 2022.

Nicolas M Müller, Piotr Kawa, Wei Herng Choong, Edresson Casanova, Eren Gölge, Thorsten Müller, Piotr Syga, Philip Sperl, and Konstantin Böttinger. Mlaad: The multi-language audio anti-spoofing dataset. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2024.

Mikhail Pautov, Nurislam Tursynbek, Marina Munkhoeva, Nikita Muravev, Aleksandr Petiushko, and Ivan Oseledets. Cc-cert: A probabilistic approach to certify general robustness of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7975–7983, 2022.

Mark E Payton. Confidence intervals for the coefficient of variation. *Kansas State University*, 1996.

Mouna Rabhi, Spiridon Bakiras, and Roberto Di Pietro. Audio-deepfake detection: Adversarial attacks and countermeasures. *Expert Systems with Applications*, 250:123941, 2024.

Mirco Ravanelli and Yoshua Bengio. Speaker recognition from raw waveform with sincnet. In *2018 IEEE spoken language technology workshop (SLT)*, pp. 1021–1028. IEEE, 2018.

Vikash Sehwag, Arjun Nitin Bhagoji, Liwei Song, Chawin Sitawarin, Daniel Cullina, Mung Chiang, and Prateek Mittal. Analyzing the robustness of open-world machine learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pp. 105–116, 2019.

Silero Team. Silero Models: pre-trained text-to-speech models made embarrassingly simple. GitHub repository, 2025. URL `https://github.com/snakers4/silero-models`.

Peeyush Singhal, Rahee Walambe, Sheela Ramanna, and Ketan Kotecha. Domain adaptation: Challenges, methods, datasets, and applications. *IEEE Access*, 11:6973–7020, 2023. doi: 10.1109/ACCESS.2023.3237025.

David Snyder, Guoguo Chen, and Daniel Povey. Musan: A music, speech, and noise corpus. *arXiv preprint arXiv:1510.08484*, 2015.

C Szegedy. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Hemlata Tak, Massimiliano Todisco, Xin Wang, Jee-weon Jung, Junichi Yamagishi, and Nicholas Evans. Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation. *arXiv preprint arXiv:2202.12233*, 2022.

Karim Tit, Teddy Furon, and Mathias Rousset. Efficient statistical assessment of neural network corruption robustness. *Advances in Neural Information Processing Systems*, 34:9253–9263, 2021.

Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee. Asvspoof 2019: Future horizons in spoofed and fake audio detection. *arXiv preprint arXiv:1904.05441*, 2019.

Kun Wang, Meng Chen, Li Lu, Jingwen Feng, Qianniu Chen, Zhongjie Ba, Kui Ren, and Chun Chen. From one stolen utterance: Assessing the risks of voice cloning in the aigc era. In *2025 IEEE Symposium on Security and Privacy (SP)*, pp. 4663–4681. IEEE, 2025.

Xin Wang, Junichi Yamagishi, Massimiliano Todisco, Héctor Delgado, Andreas Nautsch, Nicholas Evans, Md Sahidullah, Ville Vestman, Tomi Kinnunen, Kong Aik Lee, et al. Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech. *Computer Speech & Language*, 64:101114, 2020.

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-weon Jung, Hye-jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi Kinnunen, et al. Asvspoof 5: Crowdsourced speech data, deepfakes, and adversarial attacks at scale. *arXiv preprint arXiv:2408.08739*, 2024.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

Yihan Wu, Jee-Weon Jung, Hye-Jin Shim, Xin Cheng, and Xin Wang. Wildspoof: advancing in-the-wild data in text-to-speech generation and spoofing-aware automatic speaker verification. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21922–21924. IEEE, 2026.

Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, et al. Asvspoof 2021: accelerating progress in spoofed and deepfake speech detection. *arXiv preprint arXiv:2109.00537*, 2021.

Jiangyan Yi, Ruibo Fu, Jianhua Tao, Shuai Nie, Haoxin Ma, Chenglong Wang, Tao Wang, Zhengkun Tian, Ye Bai, Cunhang Fan, et al. Add 2022: the first audio deep synthesis detection challenge. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 9216–9220. IEEE, 2022.

Jiangyan Yi, Jianhua Tao, Ruibo Fu, Xinrui Yan, Chenglong Wang, Tao Wang, Chu Yuan Zhang, Xiaohui Zhang, Yan Zhao, Yong Ren, et al. Add 2023: the second audio deepfake detection challenge. *arXiv preprint arXiv:2305.13774*, 2023.

A Additional results

A.1 Full evaluation

Transformation parameter spaces. Table 5 summarizes the transformation families and parameter ranges used in the robustness verification experiments. The transformation parameters are sampled from the corresponding parameter spaces, while RIR perturbations are generated using impulse responses from the OpenSLR dataset. The composite transformation combines gain, low-pass filtering, and additive Gaussian noise.

Table 4: Robustness verification results against parametric input transformations at nominal failure probability $\alpha = 10^{-6}$. All reported metrics are percentages, evaluated on the 298 initially correctly classified audios out of 300; the PCA thresholds ε are probabilities. The total sampling budget per audio is m. CCC: CC-Cert; EBI: Empirical Bernstein's inequality; BCI: Bernoulli–Cantelli inequality; AVU: Adaptive Voice Antispoofing error upperbounding.

Transform	Method	m	Mean (%)				PCA (%)			
			$\mathcal{A}(x)$	$r(x)$	$\widehat{p}(x)$	Cost	$\varepsilon = 10^{-5}$	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-2}$	$\varepsilon = 0.05$
LPF	CCC	20000	0.133	0.093	0.040	0.088	98.322	98.993	99.664	99.664
	EBI		0.213	0.173	0.040	0.168	0.000	0.000	99.664	99.664
	BCI		0.194	0.155	0.040	0.149	0.000	0.000	99.664	99.664
	AVU		0.046	0.006	0.040	0.001	98.322	98.993	99.664	99.664
HPF	CCC	20000	10.114	2.765	7.349	2.622	85.570	87.919	88.591	88.926
	EBI		7.640	0.291	7.350	0.149	0.000	0.000	88.926	88.926
	BCI		7.734	0.384	7.350	0.242	0.000	0.000	88.926	88.926
	AVU		7.501	0.152	7.349	0.009	85.570	87.248	88.926	88.926
Background noise	CCC	6000	18.481	16.878	1.603	16.256	51.007	53.356	55.705	60.067
	EBI		2.535	0.930	1.604	0.309	0.000	0.000	71.477	88.255
	BCI		2.791	1.186	1.604	0.566	0.000	0.000	67.785	87.248
	AVU		8.750	7.145	1.605	6.525	50.671	52.685	58.725	79.530
Pitch Shift	CCC	20000	33.789	12.314	21.475	11.755	59.060	61.745	62.081	62.752
	EBI		22.138	0.668	21.470	0.104	0.000	0.000	63.423	63.423
	BCI		22.551	1.081	21.470	0.517	0.000	0.000	63.423	63.423
	AVU		22.075	0.598	21.477	0.041	59.060	61.745	63.423	63.423
BPF (1)	CCC	20000	20.964	12.733	8.231	12.202	50.336	52.349	55.369	61.409
	EBI		8.851	0.621	8.229	0.089	0.000	0.000	60.067	72.483
	BCI		9.214	0.985	8.229	0.452	0.000	0.000	59.396	71.141
	AVU		8.801	0.571	8.230	0.039	49.664	50.336	60.067	72.483
BPF (2)	CCC	20000	30.984	21.450	9.534	20.618	12.416	16.107	21.812	29.866
	EBI		10.393	0.856	9.538	0.028	0.000	0.000	35.235	62.081
	BCI		10.969	1.431	9.538	0.603	0.000	0.000	31.208	60.403
	AVU		10.437	0.904	9.534	0.071	12.752	14.765	34.564	62.081
Gain (1)	CCC	20000	0.094	0.063	0.031	0.056	99.329	99.329	99.329	99.329
	EBI		0.206	0.175	0.031	0.168	0.000	0.000	99.329	99.664
	BCI		0.193	0.161	0.031	0.155	0.000	0.000	99.329	99.329
	AVU		0.038	0.007	0.031	0.000	99.329	99.329	99.329	99.664
Gain (2)	CCC	20000	2.818	1.995	0.823	1.921	92.282	94.966	94.966	95.302
	EBI		1.056	0.234	0.822	0.160	0.000	0.000	94.966	95.638
	BCI		1.100	0.278	0.822	0.204	0.000	0.000	94.966	95.638
	AVU		0.902	0.080	0.822	0.006	92.282	93.960	94.966	95.638
RIR	CCC	20000	22.659	15.701	6.958	15.087	41.611	45.302	47.315	51.007
	EBI		7.648	0.691	6.958	0.077	0.000	0.000	51.678	69.799
	BCI		8.051	1.093	6.958	0.479	0.000	0.000	50.336	67.114
	AVU		7.617	0.656	6.961	0.045	41.611	43.289	51.007	69.463
Composite	CCC	20000	12.893	8.936	3.957	8.665	66.443	67.785	70.470	75.168
	EBI		4.343	0.384	3.959	0.115	0.000	0.000	82.886	87.919
	BCI		6.098	2.139	3.959	1.870	0.000	0.000	79.530	83.893
	AVU		6.202	2.244	3.958	1.974	65.436	66.779	79.195	84.899

A.2 TTS VERIFICATION CONDITION

A.3 OPTIMAL VALUES OF HYPERPARAMETERS

For the CC-Cert backbone, in Figures 3, 5, 7, 2, we report the dependency of PCA on the values of augmentation budget m and its distribution. In Table 6, we report averaged probability bounds from

Table 5: Parameter spaces of the transformations used for robustness verification.

Transformation	Parameter space Θ
LPF	$\omega_{\max} \in [2.5, 3]$ kHz
HPF	$\omega_{\min} \in [0.5, 1]$ kHz
Background noise	SNR $\in [15, 30]$ dB
Pitch shift	ST $\in [-6, 6]$
BPF (1)	$\omega \in [200, 1500], \quad b \in [1.2, 1.5]$
BPF (2)	$\omega \in [200, 4000], \quad b \in [0.5, 1.99]$
Gain (1)	$\gamma \in [-10, 10]$ dB
Gain (2)	$\gamma \in [-10, 20]$ dB
RIR	$r \in$ OpenSLR dataset
Composite	$\gamma \in [-10, 10]$ dB, $\omega_{\max} \in [2.5, 3]$ kHz, $\sigma \in [0.01, 0.03]$

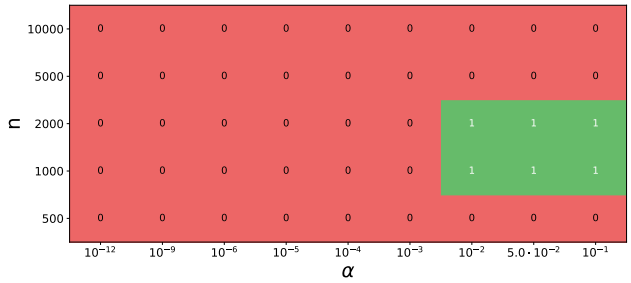

Figure 1: Verification condition result from Eq. 28 for the pre-finetuned f against Vosk TTS vs. various α and n given fixed $m = 130000$, $\delta = 0.9$, and $\varepsilon = 0.1$. CC-Cert backbone.

Algorithm 1 along with the averaged error probability $\overline{p}(x)$ for specific transforms. In Figures 4 and 6, we compare PCA for the different values of α given fixed (m, n, k).

Recall that the verification method yields an upper bound for the probability of misclassification of the audio subject to the transformation from Eq. 6. This upper bound may be over-conservative, i.e., significantly higher than the actual unknown value of $\mathbb{P}_{\theta \sim \Theta}[h(x) \neq h(x')]$. We also highlight that the distribution of the augmentation budget (namely, $m = n \times k$) into k chunks of n samples each affects the upper bound for error probability (see, for example, Eq. 10). To study these dependencies, we conduct additional experiments to illustrate the effect of the computational budget re-distribution.

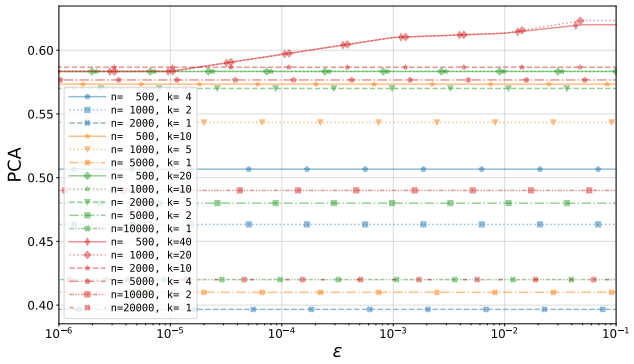

Figure 2: Dependence of PCA on (m, n, k) for the pitch shift transform, ST $\in [-6, 6]$ semitones. The confidence level is set to be $\alpha = 10^{-6}$. Curves sharing the same color correspond to the same augmentation budget m, while line styles and marker types indicate variations in n and k, respectively.

From the results, we make several observations. Firstly, the source model show strong robustness against LPF, HPF, and time stretch transforms, demonstrating low average error probability $\overline{p}$; sec-

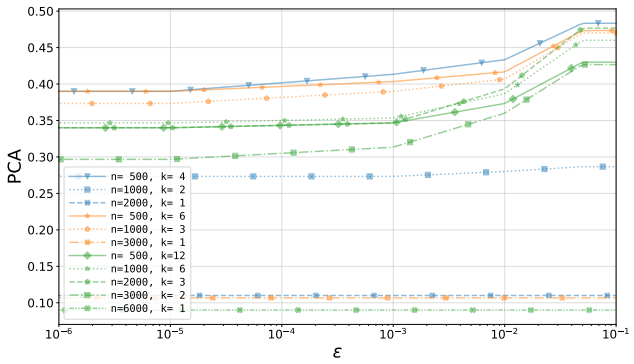

Figure 3: Dependence of PCA on (m, n, k) for background noise perturbations with $\mathrm{SNR} \in [15, 30]$. The confidence level is set to $\alpha = 10^{-6}$. Curves sharing the same color correspond to an identical computational budget m, while line styles and marker types indicate variations in n and k, respectively.

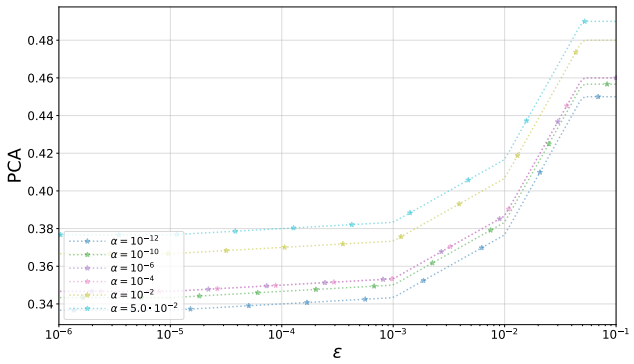

Figure 4: Dependence of PCA on α for background noise perturbations with $\mathrm{SNR} \in [15, 30]$. $m = 6000,\ n = 1000,\ k = 6$ are fixed.

ondly, the different distributions of the total augmentation budget in the form $m = n \times k$ into k splits of n samples each (see, for example, Eq. 8) do not always affect the verification results, but can be used to control the average error probability; finally, we highlight that the broader the parameters'

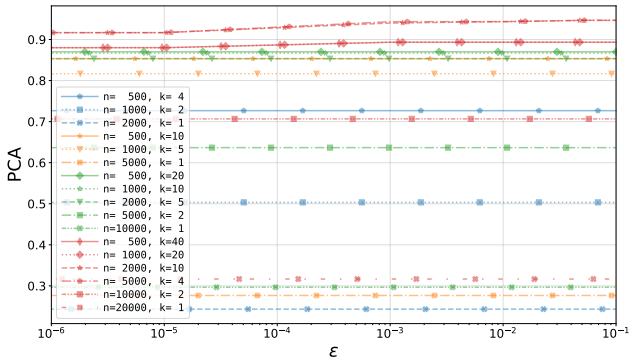

Figure 5: Dependence of PCA on (m, n, k) for the gain adjustment transform with $\gamma \in [-10, 20]$ dB. The confidence level is set to $\alpha = 10^{-6}$. Curves sharing the same color correspond to the same augmentation budget m, while line styles and marker types indicate variations in n and k, respectively.

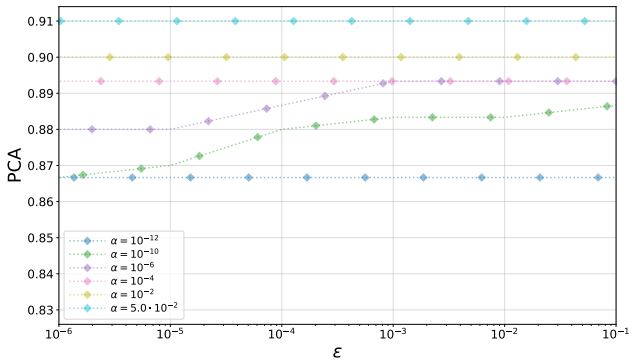

Figure 6: Dependence of PCA on α for the gain adjustment transform with $\gamma \in [-10, 20]$ dB. The values $m = 20000,\ n = 1000,\ k = 20$ are fixed.

space Θ for the given transform, the worse the expected robustness of the model to that transform is (see, for instance, the verification results for BPF for different Θ).

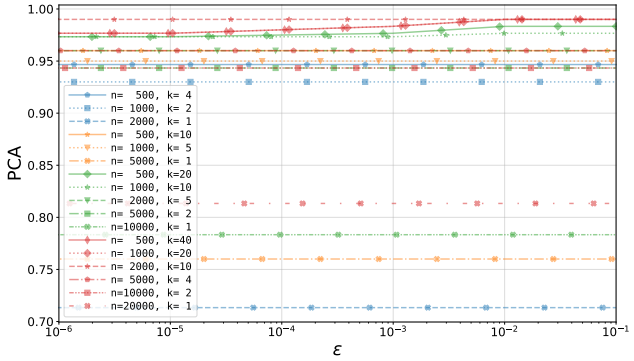

Figure 7: Dependence of PCA on (m, n, k) for the low pass filter with the cutoff frequency ω_{max} is randomly sampled from $[2500, 3000]$ Hz range. The confidence level is set to $\alpha = 10^{-6}$. Curves sharing the same color correspond to the same augmentation budget m, while line styles and marker types indicate variations in n and k, respectively.

Namely, for the CC-Cert, given a fixed budget m, we compute the value of PCA and error probability for different pairs (n_i, k_i): $n_i \times k_i = m$. We present the PCA results for LPF in Fig. 7 and Table 6, for the background noise in Fig. 3, gain in Fig. 5, and pitch shift in Fig. 2. It can be observed that not only does a budget impact PCA. For instance, $k = 1$ uniformly yields the worst PCA results, and generally increasing k improves verification results for a fixed m, as it improves values of $p(x)$. However, it is not always satisfied, for example, see Fig. 3. To obtain a tighter bound, one would be recommended to balance the augmentation budget towards increasing the value of k.

In Figures 4 and 6, the dependency of verification results on the confidence level α for background noise and gain transforms is illustrated. Generally, a higher value of α leads to less strict verification conditions and larger PCA. Additionally, for the TTS, we studied the dependency of PCA on values of n and α, with the other parameters fixed. The overall verification results are shown in Fig. 1. It is worth mentioning that the lower the value of n, the lower the bound on the error probability p, but the looser the algorithm output $\mathcal{A}$. To obtain a tighter verification results, a balance between the values of p and $\mathcal{A}$ should be found.

Table 6: Dependency of average probability of error $\overline{p}$ for different values (n, k, m), low pass filter, $\Theta = [2500,\ 3000]$ Hz. The confidence level is set to $\alpha = 10^{-6}$.

n	k	m	$\overline{\mathcal{A}(x)}$	$\overline{p(x)}$
500	4	2000	$4.7 \cdot 10^{-3}$	$1.3 \cdot 10^{-3}$
1000	2	2000	$4.8 \cdot 10^{-3}$	$2.3 \cdot 10^{-3}$
2000	1	2000	$4.6 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$
500	10	5000	$4.8 \cdot 10^{-3}$	$2.9 \cdot 10^{-5}$
1000	5	5000	$4.7 \cdot 10^{-3}$	$7.2 \cdot 10^{-5}$
5000	1	5000	$4.7 \cdot 10^{-3}$	$2.1 \cdot 10^{-3}$
500	20	10000	$4.9 \cdot 10^{-3}$	$7.6 \cdot 10^{-8}$
1000	10	10000	$4.8 \cdot 10^{-3}$	$3.8 \cdot 10^{-7}$
2000	5	10000	$4.7 \cdot 10^{-3}$	$4.3 \cdot 10^{-6}$
5000	2	10000	$4.7 \cdot 10^{-3}$	$1.2 \cdot 10^{-4}$
10000	1	10000	$4.7 \cdot 10^{-3}$	$1.0 \cdot 10^{-3}$
500	40	20000	$4.9 \cdot 10^{-3}$	$8.3 \cdot 10^{-13}$
1000	20	20000	$4.8 \cdot 10^{-3}$	$2.0 \cdot 10^{-11}$
2000	10	20000	$4.8 \cdot 10^{-3}$	$1.8 \cdot 10^{-9}$
5000	4	20000	$4.7 \cdot 10^{-3}$	$7.2 \cdot 10^{-7}$
10000	2	20000	$4.7 \cdot 10^{-3}$	$3.1 \cdot 10^{-5}$
20000	1	20000	$4.7 \cdot 10^{-3}$	$5.2 \cdot 10^{-4}$

B Additional details

B.1 Overview of conventional audio transformations

Audio transformations were implemented using `audiomentations` (Jordal et al., 2025) and `torch-audiomentations` (Jordal et al., 2022) libraries. If not stated otherwise, the default transformation parameters from the `audiomentations` library were used.

Below we list the set of audio transformations used during the training of the source model and indicate, where applicable, the range of parameters of a corresponding transformation used for augmentation of the source model during training:

- **Filters**. The cutoff frequency ω_{max} for LPF is sampled from $[3400\text{ Hz}, 7500\text{ Hz}]$; for the HPF, the cutoff frequency ω_{min} is sampled from $[100\text{ Hz}, 400\text{ Hz}]$; for the BPF, the cutoff frequency ω is sampled from $[200\text{ Hz}, 4000\text{ Hz}]$, and the bandwidth to central frequency ratio b is sampled from $[0.5, 1.99]$.
- **Pitch shift**. Pitch was randomly adjusted from -4 to 4 semitones (ST).
- **Time stretch**. A default speedup parameter $\varkappa$ was sampled from $[0.8, 1.25]$.
- **Gain**. The gain parameter γ was between -10 dB and $+20$ dB.
- **RIR**. We use the OpenSLR dataset of RIR recordings (Ko et al., 2017) and sample uniformly from it.
- **Background noise**. Similar to the RIR, we sampled from the noise subset of the Musan dataset (Snyder et al., 2015).

B.2 TTS and VC experiments

For VC and TTS experiments, from 9 to 2250 voiceprints (reference audio for VC) were used. Languages are primarily Chinese, English, and Russian.

B.3 Example algorithms

Algorithm 1 PV-VASM, CC-Cert backbone, the case of conventional input transformations

Require: Classifier f, verification dataset $\mathcal{D}$, transform ϕ and parameter space Θ, hyperparameters $n, k, d, \alpha, \varepsilon$

Ensure: Probabilistically Certified Accuracy $\mathrm{PCA}(\varepsilon, \alpha, \mathcal{D})$

$S \leftarrow 0$
for $(x, y) \in \mathcal{D}$ **do**
  $Z \in \mathbb{R}^{k \times n} \leftarrow$ AUGMENTPREDICT$(f, \phi, \Theta, x, n, k)$
  $q \leftarrow \mathbb{1}\{h(x) = y\}$
  $\mathcal{A}(x) \leftarrow \min_t \max_{j \in [k]} Y_j(Z[k])$ ▷ according to Eq. 8
  $t^* \leftarrow \arg[\min_t \max_{j \in [k]} Y_j(Z[k])]$ ▷ save the best value of t
  $\mathcal{A}(x) \leftarrow \mathcal{A}(x)/d$
  $\tilde{c} \leftarrow$ EstimateC$(Z, t^*, \alpha/2)$ ▷ one-sided interval estimation according to Eq. 13
  $p \leftarrow \left(1 + n(1-d)^2/\tilde{c}^2\right)^{-k}$
  $S \leftarrow S + \mathbb{1}\{q \wedge \mathcal{A}(x) < \varepsilon \wedge\ p < \alpha/2\}$
end for
$\mathrm{PCA}(\varepsilon, \alpha, \mathcal{D}) \leftarrow S/|\mathcal{D}|$
return $\mathrm{PCA}(\varepsilon, \alpha, \mathcal{D})$

Algorithm 2 AugmentPredict

Require: Classifier f, transform ϕ and parameter space Θ, input object x, number of samples n, number of sample means k

Ensure: The set $Z \in \mathbb{R}^{k \times n}$ of realizations of random variable Z from Eq. 7

for $j \in [k]$ **do**
  for $i \in [n]$ **do**
    $\theta \leftarrow \theta \sim \Theta, \quad x' \leftarrow \phi(x, \theta)$
    $Z[j][i] \leftarrow f(x')_2$ ▷ compute the value of p'_2 according to Eq. 4
  end for
end for
return Z