Through the Noise: Assessing Deepfake Detection Models Against Audio Distortions

Yulu Fu

Submitted in partial fulfillment of the requirements for the
Master of Music in Music Technology
in the Department of Music and Performing Arts Professions
Steinhardt School
New York University

Advisor: Dr. Brian McFee
April 21, 2025

ABSTRACT

Synthetic speech technology has advanced rapidly in recent years, making audio deepfakes increasingly convincing and accessible to create. This raises red flags for security and threatens the integrity of our digital communications. Though many studies have been done on improving detection models in clean audio environments, how detection models perform under real-world challenges is still uncertain. To address this gap, this thesis structured a series of audio augmentation to simulate common distortions in real-life such as additive noise, codec compression, time-stretching, pitch-shifting, and reverberation. To enhance the efficient testing process, a mini version of the ASVspoof 2019 dataset was generated. Then, three detection models were chosen as representatives for experiments: Wav2Vec2 (learning directly from raw audio), LCNN (learning from LFCC), and SafeEar (learning from privacy-preserving acoustic tokenization). The findings demonstrate that although Wav2Vec2 is the strongest under noise and compression, it does not work well under reverberation and extreme temporal changes. LCNN is the most stable among pitch and tempo changes but it is much more sensitive to noise. SafeEar deals with the extreme noise better than other models; however, it degenerates with downward pitch shifts and reverberation. In addition, an extended test was conducted to see if specific gaps in the acoustic frequency response might explain why all models' performance declined under reverb. The research presents a detailed comparison of model robustness under realistic conditions and offers insights on designing more reliable and adaptable audio deepfake detection systems.

.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 Motivation & Justification

Advancements in audio deepfake technologies challenge the authenticity and security of audio messages more than ever. Today, technologies can be used to create synthetic audio that is indistinguishable from the real one by using sophisticated text to speech as well as voice conversion techniques. This capacity is immensely powerful, as it makes falsification considerably more difficult to detect and challenges the very trustworthiness of audio communications. The use of synthetic media for fabricating false evidence in diverse fields can have far-reaching consequences, making it key to develop effective fraud detection strategies to combat this scourge (Dixit et al., 2023).

In response, the domain of audio deepfake detection has witnessed significant growth. Most of the research on audio deepfakes is focusing on detection models' accuracy on clean audio that comes from publicly available datasets; In comparison, the number of research that focuses on detecting deepfakes with real distortions like noise, compression, and reverberation is limited (Cohen et al., 2022). These distortions cannot be avoided as real-world applications exhibit audio transmission across lossy networks, recording in noisy or reverberant places, and modification during storage and playback. Without testing in real-world scenarios, we cannot evaluate how well the models work.

1.2 Research Question & Goals

This thesis aims to address that gap by investigating the robustness of modern audio deepfake detection models under various types of signal distortion. This study seeks to answer the following research question: What is the performance of audio deepfake detection models against real-world distortions that include additive noise, codec compression, temporal changes and reverberation?

To tackle the issue at hand, three representative detection models, namely LCNN, Wav2Vec2 and SafeEar are used to answer the question. The main goals are:

- To develop and apply a systematic augmentation pipeline that simulates real-world audio conditions through noise, compression, and reverberation;
- To compare the performance and robustness of widely used model architectures across these conditions;

- To analyze patterns of model degradation and identify strengths and weaknesses in their generalization capabilities.

Through this work, this thesis contributes a clearer picture of how current detection methods work in realistic scenarios and can help in developing more robust and reliable systems.

2. LITERATURE REVIEW

2.1 Introduction to Audio Deepfake Detection

2.1.1 History of Audio Deepfakes

The term “deepfake” initially came to the forefront of public attention in 2017 as referring to videos and images generated by deep learning algorithms (Chesney & Citron, 2019). Not long after that, researchers in the audio field started to use the approaches for processing audio, using deep learning techniques to mimic human speech, even singing now.

Some of the earliest breakthroughs in synthetic speech include WaveNet, which is created by Google DeepMind. WaveNet is a model that can synthesize natural-sounding speech directly from raw audio waveforms (van den Oord et al., 2016). This breakthrough led to even more sophisticated systems, including Tacotron and Tacotron 2 (Wang et al., 2017; Shen et al., 2018), that allow us to even be expressive and experiment with different timbres when synthesizing speech.

During the same period, voice conversion (VC) techniques also started to develop. In contrast to text-to-speech (TTS) that synthesizes speech from text, VC seeks to convert one talker's voice to another's without necessarily requiring parallel data. Techniques such as CycleGAN-VC and StarGAN-VC illustrated how adversarial training and domain adaptation can facilitate high-quality voice transfers with unpaired data (Kaneko & Kameoka, 2018; Kameoka et al., 2018).

These developments led to what we know as audio deepfakes today. While these tools can benefit the society by improving accessibility and entertainment, they also introduce risks in areas such as impersonation and disinformation. Therefore, to monitor and prevent the abuse of these technologies, it is important not only to develop detection methods, but also to test their effectiveness in realistic conditions, which is the core idea of this thesis.

2.1.2 Challenges and Real-World Risks

As high-quality deepfake tools become easier to access with limited monitoring, real problems have been raised. Audio deepfakes have already been used in things like misinformation campaigns, identity theft, and faking the voices of public figures (Dixit et al., 2023; Yasur et al., 2023). Moreover, Mirsky and Lee (2021) has listed social

engineering attacks as one potential use for synthetic audio. These attacks can involve deceiving a person or other system. Similarly, Westerlund (2019) pointed out that audio deepfakes can raise ethical and social concerns, making it harder for people to trust digital media and verify whether audio content is real.

In 2019, a notable incident about audio deepfakes that resulted in real financial losses was reported in the news, when synthetic audio was used to impersonate a company executive and trick an employee into transferring a large sum of money. The case demonstrated how convincing deepfakes can lead to financial harm and undermine trust in voice-based communication (Stupp, 2019).

2.1.3 Setting the Stage for Robustness Importance in Deepfake Detection

As audio deepfakes become a more serious threat, the research towards the development of detection systems is accelerating. However, many of the existing detection models have only been trained and evaluated on clean datasets, which only reflect a limited portion of real-life use cases. In the real world, deepfake detection systems are likely to experience complex distortions, such as noise, compression artefacts, temporal changes, and reverberation that can severely degrade model performance.

Recognizing this challenge, the ASVspoof competitions have emerged as crucial benchmarks for evaluating detection systems under more realistic conditions. Both the 2019 and 2021 competitions recognized the need for robust models and introduced standardized datasets and evaluation metrics (Todisco et al., 2019; Yamagishi et al., 2021). The 2021 challenge specifically incorporated codec-induced distortions to better simulate the compression that occurs in real-world audio transmission.

Multiple research teams have also highlighted serious performance issues when detection systems face real-world audio conditions. Müller and Pizzi (2022) discovered that many detection models break down when exposed to unexpected conditions. Similarly, Khan et al. (2023) conducted a comprehensive review of spoofing countermeasures and concluded that few systems are designed with generalization in mind. Kamble et al. (2020) additionally observed that many models tend to overfit to known attack types within benchmark datasets, limiting their real-world applicability. Wang et al. (2021) also argued that robustness should be a central focus in future model

development, as current approaches are often optimized for idealized conditions and struggle in more challenging environments. Building on these concerns, Müller emphasized that generalization remains a key weakness, particularly when models are tested on unseen synthesis methods or distortion types.(Müller, Czempin, et al., 2022).

This growing body of research highlights a major gap: although detection accuracy is improving in controlled settings, the lack of robustness continues to limit practical deployment (Müller et al., 2022; Yamagishi et al., 2021; Khan et al., 2023). This limitation drives the core focus of this research, which systematically evaluates the performance of three representative detection models—LCNN, Wav2Vec2, and SafeEar—under a range of realistic audio distortions. While LCNN and Wav2Vec2 are widely adopted in current detection pipelines (Lavrentyeva et al., 2019; Baevski et al., 2020), SafeEar represents a new and emerging approach that prioritizes audio content privacy-preserving mechanisms (Li et al., 2024b). Although model architecture like SafeEar is not widely used yet, it reflects an important direction for the future of audio deepfake detection systems, where protecting the privacy of the speech content is essential during detection.

2.2 Datasets for Audio Deepfake Detection

High-quality and well-structured datasets are essential for training and evaluating audio deepfake detection systems. Among the pool of dates available in this field, the datasets from the ASVspoof 2019 and ASVspoof 2021 challenges are one of the most popular benchmarks in the community because they provide standardised protocols and various attacks to ensure a fair and meaningful evaluation (Todisco et al., 2019; Yamagishi et al., 2021).

2.2.1 ASVspoof 2019 Logical Access (LA) Dataset

The ASVspoof 2019 collection has two categories: Logical Access (LA) and Physical Access (PA). While the PA category tackles replay attacks (where genuine recordings are played back to trick verification systems), this research focuses exclusively on the LA portion, which contains speech generated through TTS and VC (Todisco et al., 2019).

This LA dataset contains over 25,000 audio samples drawn from the VCTK corpus, featuring 107 different speakers. The spoofed samples are generated using six

attack types labeled A01 to A06, including both conventional waveform concatenation and modern neural network-based generation methods (Yamagishi et al., 2019). These attacks cover a wide range of synthesis characteristics, providing a rich basis for developing and testing detection models. Same as other datasets used in machine learning, the LA dataset is divided into training, development, and evaluation partitions with predefined protocols.

To work within practical computational limits, this study created a mini version from the development portion of the ASVspoof 2019 LA dataset. This condensed version addresses several practical challenges in the original collection, including class imbalance and excessive file volume. The mini dataset maintains careful balance between each attack type (A01–A06) and genuine recordings. The final collection contains 17,836 audio files—comprehensive enough to represent the problem space yet compact enough for efficient testing.

2.2.2 ASVspoof 2021 Dataset

The ASVspoof 2021 dataset extends from ASVspoof2019 by incorporating more real-world distortions, specifically adding audio compression codecs. These include formats such as G.722, Opus, and GSM, which reflect the use case in telecommunications and media transmission (Yamagishi et al., 2021). While the ASVspoof 2021 dataset is not used directly in the training or testing of this thesis, its design of distortions types has inspired the codec-based augmentations methods implemented in this work.

2.2.3 Relevance to This Study

The ASVspoof 2019 LA dataset is used in this study because it focuses on synthetic audio and it clearly organizes different types of spoofing attacks. By extracting a balanced and manageable mini version of this dataset, the study was able to apply a series of augmentation techniques—including additive noise, pitch shifting, temporal modifications, reverberation, and codec compression. This approach allows for robust evaluation of model performance across realistic conditions, addressing the limitations of prior work that primarily focuses on clean, undistorted audio.

While ASVspoof datasets offer standardized testing protocols, they have been criticized for their limited coverage of real-world distortions and their tendency to

promote overfitting to known attack types (Kamble et al., 2020). By using an augmented dataset with more diverse conditions, this study can address the limitation and aims to test model robustness beyond static conditions. Recent work by Cohen et al. (2022) further supports this direction, showing that targeted data augmentation strategies can enhance generalizability in voice anti-spoofing tasks.

2.3 Common Detection Approaches

A wide range of methods has been developed to detect audio deepfakes. These can be broadly grouped into three categories: models using pre-extracted features, end-to-end models, and privacy-preserving approaches. Though many of these methods perform well on existing datasets, they often fail to generalize under real-world conditions. This is the exact problem the thesis addresses.

2.3.1 Pre-Extracted Feature-Based Approaches

One of the most common strategies involves extracting acoustic features from speech signals before classification. These features provide a compact representation of the signal's spectral and temporal characteristics, capturing important patterns related to speaker identity, prosody, and synthesis artifacts. Popular feature types include Mel-Frequency Cepstral Coefficients (MFCCs), Linear Frequency Cepstral Coefficients (LFCCs), and Constant-Q Cepstral Coefficients (CQCCs) (Todisco et al., 2017; Lavrentyeva et al., 2019).

Beyond the cepstral features listed above, some systems use spectral centroid, spread, skewness, and kurtosis, which can highlight subtle differences between real and synthetic speech. Others incorporate prosodic features like pitch (F0), energy, and speaking rate—attributes that many synthesis methods struggle to replicate accurately (Xue et al., 2025).

More recently, deep embeddings like x-vectors (Snyder et al., 2018) and bottleneck features have been used to summarize speaker-level patterns. These embeddings are widely adopted in speaker verification and have been adapted to spoof detection due to their strong discriminative properties (Khan et al., 2023).

Although these models often perform well in clean conditions, they are vulnerable to input distortions. Background noise, reverberation, time misalignments, and audio

compression can interfere with feature stability and result in a sharp decline in classification accuracy (Kamble et al., 2020; Khan et al., 2023).

2.3.2 End-to-End Approaches

Unlike models that work with pre-processed features, end-to-end models aim to learn directly from raw audio (Baevski et al., 2020; Tak et al., 2021). These models don't need human-designed features because they automatically learn task-relevant representations during training. Therefore, this kind of models can capture nuanced patterns and dependencies within the audio signal that may be missed by manually defined features.

One good example is RawNet2, designed to detect fake audio directly from raw sound waves. This model uses a series of processing layers that gradually extract hierarchical features from the audio, learning to identify real versus synthetic speech (Tak et al., 2021). By avoiding fixed feature sets, RawNet2 and similar models can better learn the general signs of artificial speech patterns. Another common architecture is based on transformers, which model long-range dependencies in the audio. Wav2Vec2 is an iconic example of this architecture and will be discussed in the section below.

The main advantage of direct learning models is they can learn discriminative features directly from the data by themselves, much less dependency on feature selections. However, with every advantage, there is often a trade-off. End-to-end models typically need large amounts of data to prevent overfitting. There also are chances that they learn spurious correlations if the training data is not diverse enough. In addition, research shows that while powerful in ideal conditions, they often perform worse when faced with real-world issues like background noise or audio compression (Müller et al., 2022; Khan et al., 2023). Because of these limitations, even advanced end-to-end models should be tested not just on clean data, but also in real-world and challenging audio conditions.

2.3.3 Privacy-Preserving Approaches

As privacy and ethical concerns grow, some researchers have proposed models that operate on compressed or anonymized representations of speech. Instead of using raw waveforms or full-spectrum features, these models aim to detect manipulation without exposing or reconstructing the spoken content. Approaches in this category often

use tokenization, speech embeddings, or encrypted feature spaces to minimize data sensitivity while preserving classification performance. An example of this approach is SafeEar (Li et al., 2024b).

Although promising, privacy-aware detection is still an emerging area, with few models developed and has been tested under the same distortion-rich conditions used in traditional evaluations. This thesis contributes to this area by including a recent privacy-focused model and analyzing its behavior alongside conventional methods.

2.4 Details on Models Used in This Study

The previous sections have highlighted the overall approaches of current audio deepfake detection systems. In the next section, we focus on the specifics of these three chosen models: LCNN, Wav2Vec2 and SafeEar. These models were chosen because they can represent the current trend of audio deepfake detections and they use different approaches.

2.4.1 LCNN with LFCC Features

The LCNN model is a feature-based detection system that works with Linear Frequency Cepstral Coefficients (LFCC) extracted from speech signals. As the official baseline in the ASVspoof 2019 challenge, LCNN offers both efficiency and interpretability, making it a key benchmark in the field (Lavrentyeva et al., 2019).

Diving into the architecture of this specific LCNN, it consists of stacked 2D convolutional layers interleaved with Max-Feature-Map (MFM) activations, followed by two bidirectional LSTM (BLSTM) layers to capture temporal patterns. The model is relatively lightweight, containing about 270,000 parameters (ASVspoof Challenge, 2021).

Because it relies on pre-extracted cepstral features, LCNN serves in this study to assess how traditional feature-based models handle common real-world audio distortions. LFCCs have been shown to retain more detailed spectral envelope information than MFCCs, potentially improving the ability to distinguish between speakers and detect spoofed audio (Zhou et al., 2011).

2.4.2 Wav2Vec2

Wav2Vec2, created by Baevski et al. (2020), is an end-to-end, self-supervised model that learns directly from raw waveforms. It was originally developed for automatic

speech recognition, but recently it has been adapted for deepfake detection tasks. Unlike RawNet2 mentioned above, Wav2Vec 2.0 uses a transformer-based architecture that processes audio in two stages. First, its convolutional encoder extracts latent representations from raw waveforms; then, its transformer network builds contextual embeddings from those representations (Baevski et al., 2020).

Because the base model of Wav2Vec2 has been trained on massive amounts of unlabeled audio, it develops versatile sound representations that work well across different applications. This makes it especially valuable for testing under new conditions. In this study, Wav2Vec2 represents the category of end-to-end learning systems and is compared with approaches that use pre-extracted features and the acoustic-tokens only method that is described below.

2.4.3 SafeEar

SafeEar (Li et al., 2024b) introduces a fresh privacy-protecting approach to audio deepfake detection. Instead of analyzing full raw waveforms or complete spectrum features directly, SafeEar uses a speech tokenizer module that converts audio into semantic tokens and acoustic tokens. After that, all the semantic tokens are thrown away, leaving acoustic tokens for process only (Li et al., 2024b). These acoustic tokens then go into a transformer-based detection model that classifies them as real or fake based on their underlying patterns and sequence structure.

This design has two key purposes. First, it allows detection to proceed without keeping or rebuilding the original speech, protecting privacy. Second, by learning from compressed and tokenized inputs, the model becomes less dependent on surface-level features that might change with recording conditions. In theory, this abstraction may also provide some resilience against common audio distortions, as the model focuses more on higher-level inconsistencies rather than lower-level acoustic details.

However, as a relatively new approach, SafeEar has not yet been widely adopted across diverse conditions. Its performance under realistic audio degradations, such as noise, reverberation, and temporal changes has not been explored.

To contribute to this emerging area, this thesis includes SafeEar in the evaluation alongside other more established models like LCNN and Wav2Vec2. By testing all three under the same distorted audio conditions, this work not only compares different model

types but also offers novel evaluations of a privacy-aware detection model under diverse audio challenges.

Together, three models represent a clear and practical sample of the main approaches used in current deepfake detection. By testing each model under a suite of real-world distortions, including noise, compression, time stretches, pitch shifts, and reverberation, this thesis evaluates their robustness, generalization capacity, and practical viability in realistic deployment scenarios. It is also one of the first to compare a token-based, privacy-preserving model under distortion with models that use raw waveforms or spectral features.

2.5 Robustness in Deepfake Detection

While detection models have improved in clean conditions, a major limitation remains: many are not tested under real-world distortions. Studies have shown that noise, reverberation, and compression can significantly reduce performance, especially for models relying on spectral features (Kamble et al., 2020; Ko et al., 2017). Yamagishi et al. (2021) and Müller et al. (2022) emphasized the importance of robustness, highlighting the performance drop when systems face unfamiliar conditions. However, few studies have systematically tested a wide range of distortions in a controlled way.

This thesis addresses that gap by applying a structured distortion pipeline to three representative models, enabling a clear comparison of their robustness and generalization across realistic audio environments.

2.6 Audio Data Augmentation Techniques

Audio data augmentation is a useful way to test and improve how well models work in different conditions. It was first used in speech recognition and speaker verification to simulate real-world environments by changing training or testing audio in controlled ways (Ko et al., 2015). These strategies have since been adopted in deepfake detection to reflect the kinds of audio variations models may face in actual use cases.

Common augmentation methods include adding background noise, applying codec compression, shifting pitch and tempo, adding reverberation, and changing dynamic range. In speech recognition, Ko et al. (2015, 2017) showed that adding noise and reverberation helps models perform better across different acoustic environments. Park et al. (2019) later introduced time and frequency masking, which encouraged the

wider use of signal-level changes. These techniques have since been adopted in speaker recognition to improve performance across various devices and recording settings.

However, in the context of deepfake detection, the use of data augmentation remains limited and often inconsistent. A recent study by Cohen et al. (2022) explored augmentation strategies for voice anti spoofing and found that certain distortions—such as reverberation and noise—can significantly affect model performance. Yet many existing studies do not apply these augmentations in a systematic or reproducible way, and very few evaluate results across multiple distortion types in a unified framework.

To address this gap, this thesis introduces a structured augmentation pipeline that includes adding noise, compressing audio with codecs, modifying pitch and tempo, and adding reverberation using real-world impulse responses. By applying these distortions in a controlled and repeatable manner, the pipeline enables a more complete and fair assessment of model reliability in real-world conditions.

2.7 Evaluation Metrics

Evaluating how well audio deepfake detection models work requires standard and easy-to-understand performance metrics. In research and industry benchmarks like ASVspoof 2019 and 2021, a consistent set of evaluation tools has been adopted to allow fair comparisons between different systems. These include the Receiver Operating Characteristic (ROC) curve, Area Under the ROC Curve (AUC), Detection Error Tradeoff (DET) curve, and Equal Error Rate (EER). Each of these metrics gives a different view of model performance, particularly in how well it balances false positives (mistaking real audio for fake) and false negatives (missing actual deepfakes).

The ROC curve is a graphical plot that demonstrates the trade-off between the true positive rate (sensitivity) and the false positive rate at various thresholds (Fawcett, 2006). AUC calculates the area under the ROC graph. According to the AUC function, the area will encapsulate the value between 0 and 1. Besides, the higher AUC score object is, the better the model is able to predict 0s and 1s. The AUC is well-utilized because it does not require the definition of a threshold and thus has applicability when constraints on the operating conditions or class distributions are not present (Davis & Goadrich, 2006).

The DET curve is a variant of the ROC curve that plots the false rejection rate against the false acceptance rate, typically on a normal deviate scale. This visualization provides a more interpretable comparison of errors in systems where small differences in error rates are meaningful (Martin et al., 1997). It is particularly common in biometric and speaker verification evaluations, including ASVspoof benchmarks.

The Equal Error Rate (EER) is a scalar metric that identifies the point on the DET curve where the false acceptance rate and false rejection rate are equal. This value is often used as a single-number summary to reflect the trade-off performance of a detection system. A lower EER indicates better balance between detecting spoofed and bonafide speech, and it is one of the primary metrics reported in spoof detection challenges (Todisco et al., 2019; Yamagishi et al., 2021).

Combined, these metrics provide a complete assessment framework which is sensitive to the accuracy of the system, and trade-offs in decisions. In this thesis, ROC, AUC, DET, and EER are used to assess each model's performance under clean and distorted conditions, in line with established benchmark practices. This allows for consistent comparisons while highlighting the models' ability to generalize across different audio environments.

3. METHOD

The purpose of this thesis is to analyze the effect of audio distortions on the robustness of the chosen three deepfake detection models. By evaluating these three representative models (LCNN, Wav2Vec2, and SafeEar) on datasets augmented with various distortions, we can identify the weaknesses of each model and further investigate approaches to improve audio deepfake detection models in real world circumstances.

To achieve this goal, the methodology of this thesis has been broken down into four parts. The next sections will explain the steps sequentially, following this order: mini dataset creation, data augmentation, testing model architectures, and evaluation frames.

3.1 Mini Dataset Creation

Creating a mini dataset was the foundation of this thesis in order to evaluate detection models through a series of data augmentation. The mini dataset used is derived from the original ASVspoof 2019 Logical Access (LA) dataset . The original dataset, with approximately 25,000 audio files (about 7 GB) (Yamagishi et al, 2019), was too

large to be processed efficiently for this study, causing storage and computational constraints. Also, the dataset has a significant class imbalance between bonafide labeled (real) and spoofed labeled (fake) audio samples, which could lead to biased model training and skewed evaluation results. Therefore, to overcome these issues, the mini dataset was crafted to ensure the balanceness and manageability of the experiments while retaining the diversity of the original dataset.

To be specific, the mini dataset is built from the ASVspoof2019_LA_dev subset with the LA dataset, following the original protocol guidelines. Inside the mini set, seven groups of audio samples were identified: one for bonafide samples and six for spoofing attack types (A01–A06). Based on the information from the ASVspoof2019 evaluation plan, A01 to A03 are generated from neural network-based text-to-speech (TTS) systems, A04 is generated from a waveform concatenation TTS method, and A05 to A06 are voice conversion attacks (Yamagishi et al., 2019). To ensure class balance and perform data cleaning within the mini dataset, redundant files and unlabeled samples were removed. As a result, the mini dataset has 17836 audio files in total (about 2.3 GB), 2548 files for each type. A new protocol was also generated for the mini dataset following this structure: [Speaker_ID] [Audio_file_ID] - [Attack_type] [Label for bonafide or spoof].

By reducing the size of the original dataset while retaining its diversity, the mini dataset is able to provide a more manageable but representative base for the following data augmentation and model evaluation.

3.2 Data Augmentation

The second stage of the methodology is to apply data augmentation techniques on this mini dataset to mimic real-world audio distortions. These augmentations aim to challenge the deepfake detection models by introducing controlled modifications that replicate various environmental and technical conditions. For the thesis, I applied four types of data augmentation: noise-based augmentation, codec compression, temporal distortions, and reverb. These augmentations were intentionally chosen to expose the models to complex conditions, testing their robustness and generalizability.

3.2.1 Noise-Based Augmentation

This thesis evaluated model robustness under realistic audio degradation through the application of noise-based augmentation. Rather than using generic noise models,

three spectrally distinct noise types were carefully selected: white noise, pink noise, and brown noise. These specific noise profiles were chosen because they represent fundamentally different acoustic phenomena found in real-world recording environments (Zhang et al., 2025; Li et al., 2024b). The spectral characteristics of each noise type are as follows:

- White noise exhibits a flat frequency spectrum across all bands, effectively emulating broadband background interference such as electronic static or recording hiss.
- Pink noise demonstrates a -3 dB per octave spectral roll-off, emphasizing lower frequencies with equal energy distribution per octave, thus resembling naturally occurring ambient sounds including rainfall or wind.
- Brown noise (alternatively termed red noise) presents a steeper -6 dB per octave slope with pronounced low-frequency dominance, creating deep, rumbling background effects similar to urban environments.

Although standard practice often employs only three or five SNR levels, this research deliberately utilized seven distinct signal-to-noise ratio levels: 0 dB, 5 dB, 10 dB, 15 dB, 20 dB, 25 dB, and 30 dB SNR. This expanded range proved essential for capturing subtle performance variations that emerge between moderate and high-quality audio conditions. The experimental design thereby established a comprehensive degradation spectrum from severely compromised to near-pristine audio quality.

The implementation of white noise utilized NumPy's np.random.normal() function due to its reliable generation of zero-mean distributions with statistically consistent properties. After initial testing revealed inconsistent amplitude scaling, a specialized calculate_snr() function was developed to precisely adjust noise amplitude based on measured signal-to-noise power ratios. This approach ensured experimental consistency across all test samples despite their varying dynamic ranges.

Pink noise generation presented unique challenges regarding spectral accuracy. The implementation employed scipy.signal.lfilter with carefully calibrated coefficients to achieve the theoretical -3 dB per octave spectral slope. Despite initial filter instability at very low frequencies, normalization and proper scaling ultimately produced the desired spectral characteristics resembling natural ambient soundscapes.

For brown noise synthesis, a cumulative summation approach transformed white noise into the required spectral profile with dominant low-frequency content. This method initially introduced problematic DC offset artifacts, particularly noticeable at higher SNR levels. Additional processing addressed this issue through DC component removal and proper normalization before final scaling and signal addition, resulting in realistic low-frequency environmental noise simulation.

3.2.2 Codec Compression

Codec compression was applied as a data augmentation technique to evaluate the robustness of deepfake detection models under distortions commonly introduced during audio compression. Audio codecs are widely used in telecommunication systems, streaming platforms, and digital storage to reduce file sizes and bandwidth requirements. However, this process often introduces artifacts, such as spectral smearing, quantization noise, and reduced dynamic range, which can impact the performance of audio-based machine learning models.

The codec augmentation in this study was motivated by the compression formats featured in the ASVspoof 2021 dataset, which represent common telephony and streaming use cases (Yamagishi et al., 2021). Six codecs were implemented: A-law, μ-law, G.722, GSM, Opus, and MP3, along with a no-codec baseline to serve as a control. Each codec was chosen for its distinct compression method and practical relevance. For example, A-law and μ-law are used in traditional PSTN systems, G.722 and GSM are common in low-bandwidth speech codecs, while Opus and MP3 represent modern codecs used in VoIP and streaming.

The codec augmentation was implemented using FFmpeg for all formats, with custom encoding and decoding workflows. For most codecs, audio was first saved as a WAV file, then passed through FFmpeg with the appropriate codec-specific parameters. In the case of GSM, the signal was first resampled to 8 kHz to meet codec requirements and processed using the libopencore_amrnb codec to ensure compatibility. Additionally, an alternative GSM implementation using the python-gsm library was used, processing the signal in 160-samples frames.

By applying these codecs, this study systematically examines the ability of deepfake detection models to generalize and perform reliably under compression-induced distortions, providing insights into their robustness in real-world audio applications.

3.2.3 Temporal Distortions

Temporal distortions were applied as a data augmentation technique to simulate real-world variations in audio playback and recording. These distortions mimic scenarios such as changes in playback speed or pitch due to transmission errors, device variability, or intentional modifications. By introducing controlled temporal distortions, this study evaluates the robustness of deepfake detection models under these conditions. Two key types of temporal distortions were implemented: time-stretching and pitch-shifting.

3.2.3.1 Time-Stretching

Time-stretching involves altering the playback speed of an audio file without changing its pitch. This augmentation was implemented using the SoX (Sound eXchange) command-line audio processing tool, which enables high-quality and efficient manipulation of audio speed with consistent output fidelity (SoX Development Team, 2024).

To evaluate a wide range of playback rate variations, a total of 19 different stretch factors were applied:

- Stretch Slower (Tempo < 1.0):
  0.9×, 0.8×, 0.7×, 0.6×, 0.5×, 0.4×, 0.3×, 0.2×, 0.1×
  These simulate playback slowdowns ranging from slight lag to extreme deceleration, which can happen due to synchronization issues, malfunctioning devices, or intentional tampering.
- Stretch Faster (Tempo > 1.0):
  1.1×, 1.2×, 1.3×, 1.4×, 1.5×, 1.6×, 1.7×, 1.8×, 1.9×, 2.0×
  These reflect accelerated playback due to clock drift, compression speed-up, or other transmission-level inconsistencies.

All transformations were performed using the tempo effect in SoX, which preserves the pitch of the original audio, ensuring only the temporal characteristics were altered. This method simulates distortions commonly found in online streaming platforms, mobile voice messaging systems, and degraded communication channels.

3.2.3.2 Pitch-Shifting

Pitch-shifting involves altering the pitch of the audio signal while maintaining its original duration. This was also implemented by the SoX utility, using its pitch effect, which allows precise control of semitone-based frequency shifts without affecting timing (SoX Development Team, 2024). To simulate a wide range of spectral distortions, a total of 20 pitch-shift conditions were used:

- Pitch Shift Up (Higher Pitch):

  +1 to +10 semitones (in 1-semitone increments)

  These represent pitch increases caused by hardware tuning offsets, audio synthesis errors, or adversarial transformations.
- Pitch Shift Down (Lower Pitch):

  –1 to –10 semitones (in 1-semitone increments)

  These reflect pitch decreases resulting from hardware inconsistencies, analog playback variations, or adversarial editing.

The SoX-based pitch shifting preserves the audio duration while modifying the frequency content, allowing the models to be tested against isolated spectral distortions. These conditions are relevant in real-world scenarios where encoding settings vary across devices or content is deliberately manipulated to evade detection.

By introducing both time-stretching and pitch-shifting, this thesis provides a comprehensive analysis of how temporal distortions impact the performance of deepfake detection models. These augmentations simulate common real-world challenges, enabling a rigorous evaluation of model robustness and generalizability.

3.2.4 Reverberation

To simulate the impact of room acoustics on the performance of deepfake detection models, reverberation was applied to the audio dataset using a convolution-based method. Reverberation is a common environmental distortion in real-world recordings, caused by the reflections of sound waves in physical spaces. Therefore, the effect of reverb is essential to experiment on in order for audio deepfake detection models to be prepared for daily uses.

This study implemented reverberation augmentation by convolving the original clean audio files with real-world impulse responses (IRs) sourced from three spatial

categories: large rooms, small rooms, and open spaces. There are three IRs within each category, selected from environments that are representative:

- Small Rooms: Bathroom, Car Interior, Elevator Interior — all obtained from Freesound.org (Freesound, 2012).
- Large Rooms: Baptist Church and Elveden Hall from the OpenAIR library (Murphy & Shelley, 2010), and an Underground Train Station from Freesound.org.
- Open Spaces: Koli National Park, Abies Grandis Forest, and Haven Beach (Puerto Rico) — all from Freesound.org.

Overall, these nine IRs were chosen to represent various reverberation characteristics.

Before using these IRs, each one was analyzed to ensure it was valid and acoustically distinctive. The RT30 was approximated by converting the IR waveform into a decibel scale and visually estimating the time required for the energy to decay by 30 dB SNR. This revealed a range of decay times, from under 0.5 seconds in small rooms to over 2.5 seconds in large halls and open environments. In addition, the sample rate of the IRs were standardized to 44.1kHz and their peak amplitude was checked. The visualization of each IR were also plotted using Matplotlib, including three key views: the time-domain waveform, the decibel-scale envelope to highlight reverberation decay, and the frequency spectrum derived from the magnitude of the FFT. These plots aided in verifying the reverberation behavior and spectral characteristics of each IR, further facilitating the understanding of how reverb affects the performance of deepfake detection models.

After choosing the IRs, these IRs were convolved with the original clean mini dataset using scipy.signal.fftconvolve. A wet gain of 0.7 and a dry mix of 0.3 were applied to maintain clarity while making the reverberation effect perceptible. The processed signals were normalized to avoid clipping, and files were saved in structured subdirectories based on IR category and environment.

3.3 Testing On Models

Testing audio deepfake detection models is the core component of this thesis. With a comprehensive set of augmented datasets designed to simulate a wide range of

real-world conditions, this section evaluates how well each model performs in detecting synthetic speech under distortion. Three detection models are selected for testing: LCNN, Wav2Vec2, and SafeEar. Each model represents a different architectural approach, as previously described in Section 2.4. While Section 2.4 covers the design and structure of these models, the focus here is on how they were trained and evaluated using the mini dataset and augmented conditions.

3.3.1 LCNN with LFCC Features

The LCNN model used in this study is the official ASVspoof2019 baseline system, which relies on Linear Frequency Cepstral Coefficients (LFCC) for input features. Rather than training a new model, this study uses the pre-trained LCNN weights provided in the ASVspoof2019 GitHub repository (ASVspoof Challenge, 2021), allowing for consistent benchmarking against a well-established baseline.

To evaluate the LCNN model on augmented datasets, the official evaluation pipeline was adapted by modifying the provided 01_wrapper_eval.sh script. In this modified version, custom loops were created to iterate over all augmented dataset variants, such as pitch-shifted and time-stretched audio. For each distortion type and condition, the script updated input paths, specified output folders, and ran the evaluation using 02_eval_alternative.sh, which computes scores for each sample using the pre-trained network (trained_network.pt). Output files and log reports were saved in a structured directory format for further analysis.

The model's internal feature extraction and inference logic were left unchanged. It is worth noting that the output raw confidence scores of LCNN are not bounded. The predictions follow the general rule that higher values indicate stronger likelihoods of bonafide speech. These scores were collected for all test conditions and used directly for calculating performance metrics including ROC, AUC, DET, and EER without additional normalization or thresholding.

3.3.2 Wav2Vec2: Transformer-Based Model

The second model used in this study is a fine-tuned version of Facebook AI's Wav2Vec2, a transformer-based architecture that learns directly from raw audio waveforms (Baevski et al., 2020). In this setup, the Wav2Vec2 base model was adapted for binary audio deepfake classification using a custom classification head. The model

consists of the pretrained Wav2Vec2 encoder followed by two linear layers with a ReLU activation and dropout in between, designed to output a bonafide or spoof prediction.

The model was fine-tuned on the ASVspoof2019 LA training set using Hugging Face's Trainer API. During training, audio files were loaded on-the-fly using the soundfile library and then preprocessed into input tensors using the Wav2Vec2FeatureExtractor. A custom padding-based data collator was used to handle variable-length audio and ensure consistent batch dimensions. To conserve GPU memory and reduce the risk of overfitting, the feature encoder within Wav2Vec2 was frozen, allowing only the classification head to be updated. The model was trained for two epochs with a batch size of 8, using a learning rate of 3e-5 and gradient accumulation over 4 steps to simulate a larger effective batch size. After training, performance was evaluated on the ASVspoof2019 LA development set, and the best-performing model checkpoint was selected based on the lowest Equal Error Rate (EER).

For inference, the trained Wav2Vec2 model was applied to each distortion condition within the augmented mini dataset. A dedicated inference script was created to automate this process by loading the trained model, extracting features from each audio file, and computing the corresponding class logits. To obtain scores consistent with the other models, the softmax probability assigned to the "spoof" class was subtracted from 1, yielding a bonafide confidence score:

```
bonafide_score = 1 - row['spoof_score']
```

This transformation made the score format consistent with LCNN and SafeEar, so all three models could be evaluated in the same way using ROC, AUC, DET, and EER metrics. The predictions were processed in batches and saved as structured CSV files for each augmentation condition to support later analysis.

3.3.3 SafeEar: Neural Codec-Based Pipeline

The third model evaluated in this study is SafeEar, a privacy-preserving audio deepfake detection system that leverages tokenized acoustic representations instead of raw waveforms or conventional spectral features. SafeEar is composed of two main components: a speech tokenizer that generates acoustic tokens using a pre-trained HuBERT model, and a detection module that classifies token sequences as either

bonafide or spoofed. This design aims to protect speaker identity by decoupling semantic content from acoustic cues.

In this study, SafeEar was trained from scratch on the ASVspoof2019 LA training set, following the authors' official implementation (Li et al., 2024a). Feature extraction was carried out using the dump_hubert_avg_feature.py script to generate averaged HuBERT Layer 9 token embeddings for each audio sample. These embeddings were used as input to the detection model, which was trained using the configuration provided in train19.yaml. Training and model orchestration were handled using PyTorch Lightning and Hydra, with all modules initialized from YAML-based configuration files.

For inference, a custom evaluation pipeline was created by modifying the official test.py script. The system was run on each distorted condition of the augmented mini dataset using consistent protocols and pretrained checkpoints. During inference, audio files were loaded and padded or truncated to a fixed length, passed through the speech tokenizer, and then classified by the detection module. Output confidence scores—unbounded real values indicating the likelihood of spoofed audio—were collected for each sample and used directly for evaluation. No score normalization or thresholding was applied.

As with the other models, the SafeEar predictions were saved in structured log and score files for each augmentation condition. These results were then analyzed using standard evaluation metrics, including ROC, AUC, DET, and EER, for consistency with LCNN and Wav2Vec2.

3.4 Evaluation Framework

All models were evaluated using the same augmented datasets introduced earlier in the thesis to ensure a balanced and thorough assessment. This standardized approach enables direct comparisons between the baseline LCNN, the transformer-based Wav2Vec2, and the neural codec-based pipeline detector, SafeEar—highlighting how each model performs under different types of audio distortion, as well as their respective strengths and limitations.

3.4.1 Implementation of Evaluation Metrics

The evaluation process is structured in two main stages: metric calculation and performance visualization. In the first stage, each model is tested against a range of

conditions, and key performance metrics are computed. These include the Receiver Operating Characteristic (ROC) curve, the Detection Error Tradeoff (DET) curve, the Area Under the Curve (AUC), and the Equal Error Rate (EER). These metrics are commonly used in the fields of speaker verification and spoof detection because they offer threshold-independent insights into model behavior. Unlike metrics such as accuracy, precision, recall, and F1-score—which require a fixed decision threshold—ROC and DET-based metrics characterize performance across the full spectrum of thresholds, offering a more complete picture of a model's capabilities.

3.4.2 Receiver Operating Characteristic (ROC) Curve and Area Under the Curve (AUC)

The ROC curve illustrates the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at various threshold levels. It provides a visual summary of how well a model distinguishes between bonafide and spoofed inputs across all possible decision boundaries, independent of any fixed threshold.

For each condition and model, the ROC curve was computed using the roc_curve() function from scikit-learn. By providing an array of ground truth binary labels (y_true) and predicted confidence scores (y_score), the TPR and FPR values were calculated across a range of thresholds.

TPR and FPR are calculated as below:

TPR = True Positives (TP) / [True Positives (TP) + False Negatives (FN)]

FPR = False Positives (FP) / [False Positives (FP) + True Negatives (TN)]

Once the ROC curve was obtained, the Area Under the Curve (AUC) was calculated using the roc_auc_score() function, which applies numerical integration using the trapezoidal rule. The AUC provides a single scalar value that quantifies the model's overall ability to rank bonafide samples higher than spoofed ones. By convention, AUC values range from 0 to 1, where values closer to 1 indicate better discrimination. A model with an AUC near 0.5 performs no better than random guessing, while a value closer to 1 reflects strong class separability.

The ROC curve and AUC are inherently linked: the ROC curve visualizes a model's performance across thresholds, while the AUC summarizes the shape of that curve into a single value. Thus, AUC directly reflects the overall quality of the ROC

curve. A higher AUC implies that the ROC curve remains closer to the top-left corner of the plot, where the true positive rate is maximized and the false positive rate is minimized.

In addition to computing ROC curves for each individual distortion condition, an aggregated ROC curve was also generated for each model to assess its overall robustness across all test scenarios. This was done by merging the prediction scores and ground truth labels from all conditions—including clean and distorted audio—into a single evaluation set per model. The resulting ROC curve reflects each model's global performance under mixed real-world conditions, rather than performance on a single controlled distortion. Aggregated ROC curves for all three models were plotted together in a single figure, enabling direct visual comparison. The corresponding AUC values serve as summary indicators of each model's general classification capability across diverse distortions.

3.4.3 Detection Error Tradeoff (DET) curve & Equal Error Rate (EER)

The Detection Error Tradeoff (DET) curve provides a visual representation of the trade-off between the false rejection rate (FRR) and the false acceptance rate (FAR) across different decision thresholds. While similar in concept to the ROC curve, the DET curve uses a normal deviate scale, which spreads out low-error regions and makes fine-grained differences in model performance easier to interpret (Scikit-Learn, 2024).

For each condition and model, the DET curve was generated using the same set of predictions used for ROC computation. The roc_curve() function from scikit-learn was used to calculate the false positive rate (FPR) and true positive rate (TPR). The false rejection rate (FRR) was derived from the TPR as:

$$\text{False Rejection Rate (FRR)} = 1 - \text{True Positive Rate (TPR)}$$

$$\text{False Acceptance Rate (FAR)} = \text{False Positive Rate (FPR)}$$

These values were then scaled to percentages and transformed to the normal deviate scale using the scipy.special.ndtri() function.

An important scalar metric derived from the DET curve is the Equal Error Rate (EER), which represents the point at which the false acceptance rate equals the false rejection rate. It offers a balanced summary of model performance when the costs of false positives and false negatives are assumed to be equal. In this framework, the EER was

computed using Brent's root-finding method (scipy.optimize.brentq) to locate the threshold where FAR and FRR intersect. The corresponding threshold at the EER point was also recorded for further analysis.

As with the ROC curves, aggregated DET curves were also generated to evaluate each model's robustness under real-world conditions by merging predictions from all distortion types—including clean and augmented conditions—into a single evaluation set per model. The resulting DET curves capture each model's overall error trade-off behavior across diverse input scenarios. Aggregated curves for all three models were plotted together with their EER values marked, enabling direct visual comparison of architecture-level performance.

4. ANALYSIS & RESULTS

4.1 Overall Performance

Using the evaluation procedures outlined in the methodology, this section offers a comparative overview of how the three models perform across various distortion types. It provides a high-level summary of each model's ability to handle a diverse range of real-world audio distortions, including noise, codec compression, temporal changes, and reverberation.

To generate the ROC and DET curves shown below, prediction scores from all individual test conditions were combined into a single evaluation set for each model. This involved merging the raw model outputs with their corresponding labels across all scenarios. Standardization was then applied to ensure consistent scoring, where higher values consistently indicate a greater likelihood that a sample is bonafide.

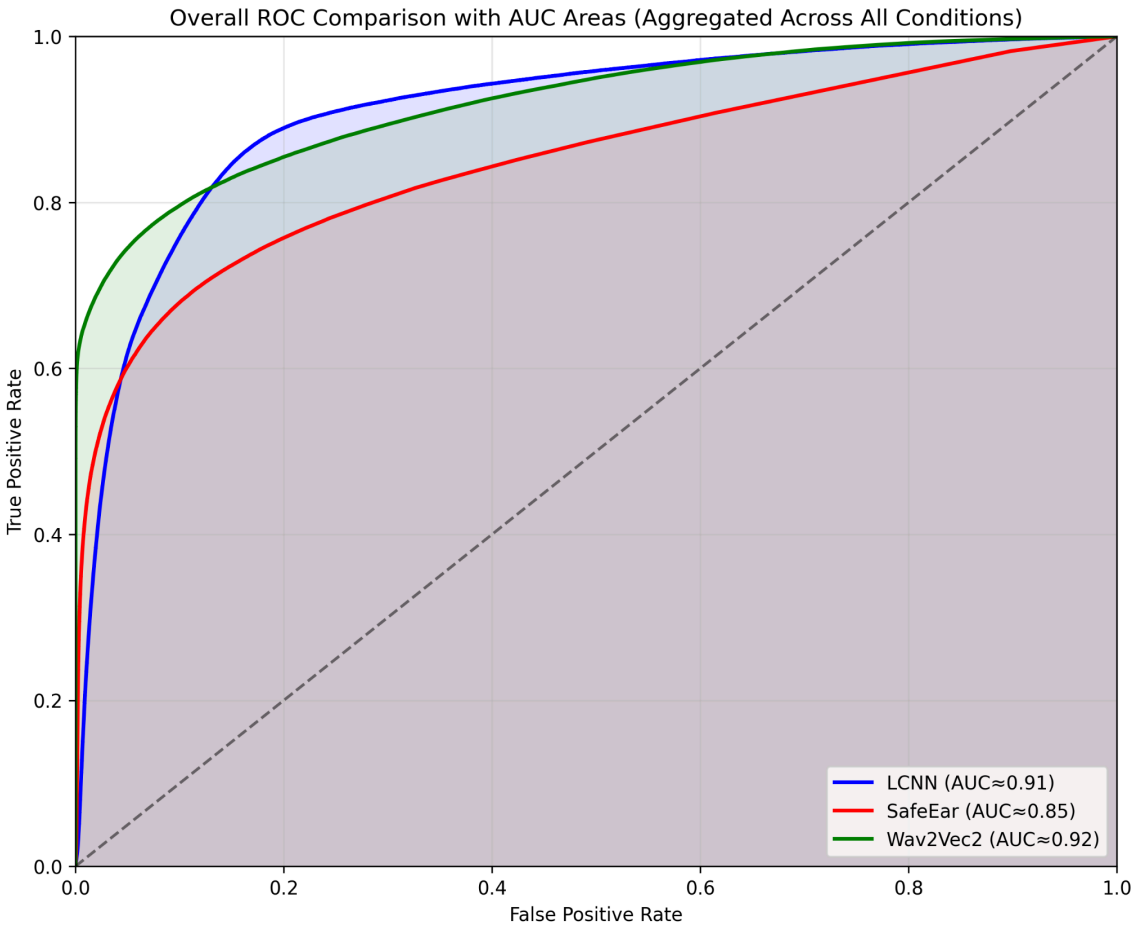

Figure 1: Overall ROC Comparison for LCNN, SafeEar, and Wav2Vec2

4.1.1 Overall ROC Curve (Figure 1)

In Figure 1, the Receiver Operating Characteristic (ROC) curves summarize the true positive and false positive tradeoff across all thresholds, with a higher curve at top left corner indicating better overall detection performance. The LCNN baseline model provided by the ASVspoof committee is labeled in blue, the neural codec-based pipeline detector named SafeEar is labeled in red, and the transformer-based Wav2cec2 model is labeled in green.

Among the three models, Wav2Vec2 demonstrates the best overall performance, with an Area Under the Curve (AUC) of approximately 0.92. Its ROC curve rises sharply from the origin, achieving a high TPR while maintaining a low FPR. This steep ascent indicates that the model can effectively distinguish between bonafide and spoofed audio with minimal false alarms.

The LCNN model, with an AUC of around 0.91, also performs well, though its curve is slightly less steep than the curve of Wav2Vec2 in the early region. It requires a higher FPR to reach the same level of TPR, especially during the initial threshold range, but still maintains high performance overall.

The SafeEar curve is generally flatter with the AUC at around 0.85, indicating weaker performance. Interestingly, in the very low FPR region (below 2%), SafeEar briefly outperforms LCNN, showing a slightly steeper initial slope. This means SafeEar starts off identifying some bonafide samples more efficiently at very low FPRs. Beyond that point, LCNN quickly surpasses it and maintains stronger performance for the rest of the curve.

These results confirm that Wav2Vec2 is the most robust model overall, followed by LCNN. While SafeEar shows potential under highly constrained conditions, it lacks consistency in more general detection scenarios.

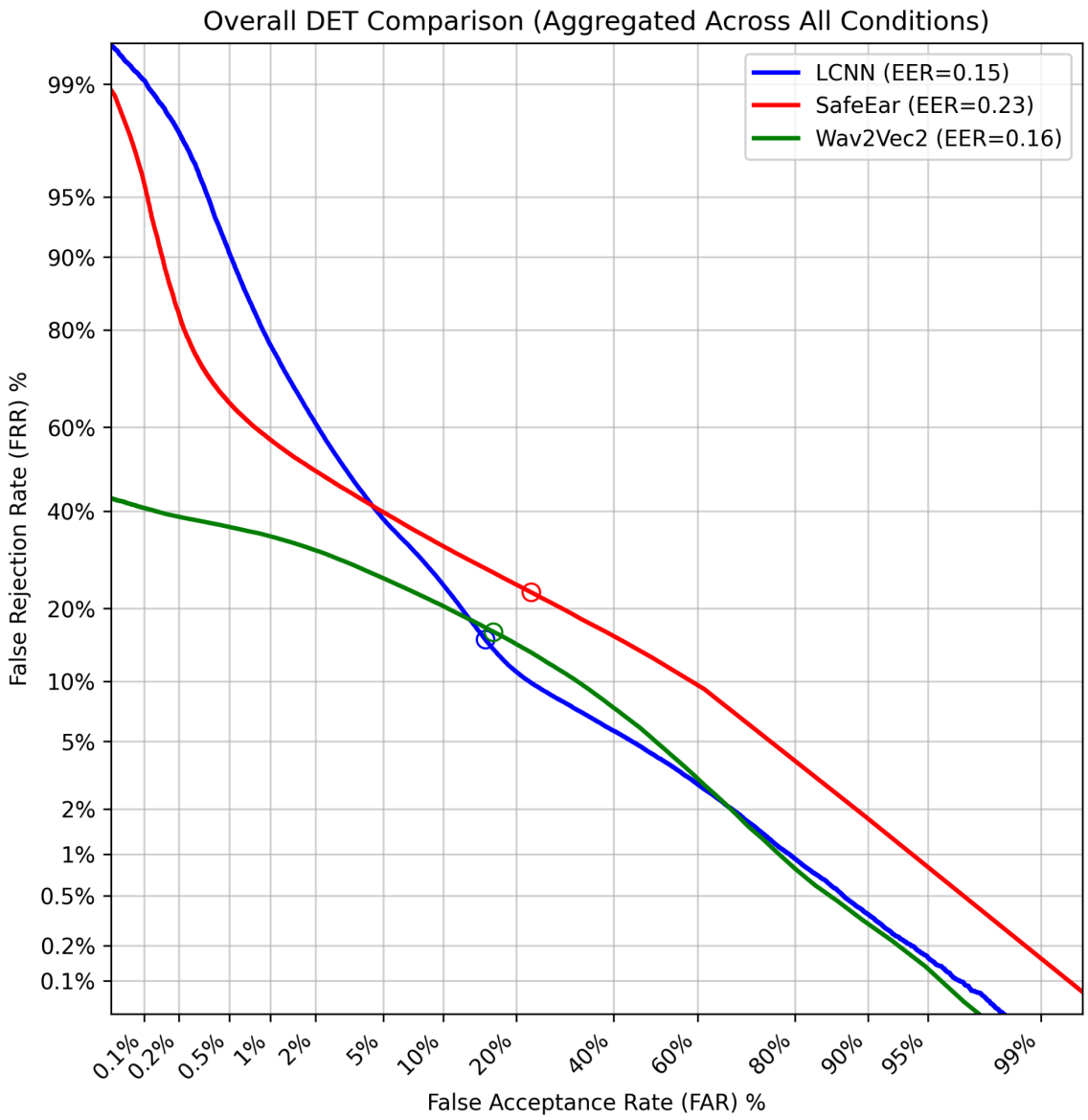

Figure 2: Overall DET Comparison for LCNN, SafeEar, and Wav2Vec2

4.1.2 DET Curve Overall (Figure 2)

Figure 2 displays the Detection Error Tradeoff (DET) curves for the LCNN, SafeEar, and Wav2Vec2 models. These curves show the relationship between the FAR and FRR, plotted on a normal deviate scale. Each curve represents how each model balances sensitivity and specificity across threshold settings.

Looking at figure 2, the Wav2Vec2 curve sits clearly below the others across most of the range. It begins with a sharp drop, reaching an FRR around 35% at the lowest FAR (0.1%), and continues declining steeply. This steep drop indicates that even a slight

relaxation of the decision threshold quickly recovers many bonafide samples while keeping spoof acceptance low. Its EER of 0.16 further confirms its strong and stable classification ability across thresholds.

LCNN also performs well, with an EER of 0.15, which is slightly lower than that of Wav2Vec2. However, this advantage in EER value can be misleading when viewed in isolation. In the early part of the DET curve, particularly at low FAR, LCNN exhibits a much higher FRR compared to Wav2Vec2. This means that LCNN struggles to correctly identify bonafide samples under strict thresholds. As FAR increases, the LCNN curve drops rapidly and eventually aligns closely with Wav2Vec2, even falling slightly below it between FAR values of approximately 15% to 60%. This suggests that LCNN becomes increasingly efficient at recovering correct detections in more lenient operating conditions.

The SafeEar appears the highest in the DET plot among the three, especially in the low-FAR region. It also declines more slowly than the other curves and never quite reaches the same low region of the plot. However, it is worth noting that the SafeEar starts lower on the DET curve than LCNN before being overtaken when FAR is just below 5% and FRR is between 40% to 60%. This suggests that SafeEar works better when the system is very strict, but its performance drops as the threshold becomes more relaxed. In general, SafeEar struggles to lower both error rates at the same time, which shows it doesn't adapt well to different testing conditions.

4.1.3 Summary for Overall ROC and DET curves

In summary, the ROC and DET curves provide consistent insights into the performance of the three models, showing a clear ranking in overall effectiveness. Wav2Vec2 performs the best under distortions overall, with the highest AUC and a consistently low DET curve across most thresholds. LCNN follows closely, with a slightly lower AUC but a marginally better EER; however, its performance is less stable under strict thresholds, where it shows higher false rejection rates. SafeEar, while briefly outperforming LCNN at very low FPRs, shows the weakest overall performance, with a flatter ROC curve and the highest DET curve across most operating points.

4.2 Noise-based Augmentation Performance

Now that the overall performance of the three models has been compared, the next step is to look at how each model handles specific types of audio distortions. This section focuses on noise-based augmentations, which involve adding different types of background noise to the audio. As mentioned in the methodology section, there are three types of noise used in augmentation: white noise, pink noise, and brown noise. Below, each type is examined to see how it affects the models' ability to detect deepfakes under varying signal-to-noise ratios (SNRs), starting with white noise.

4.2.1 White Noise

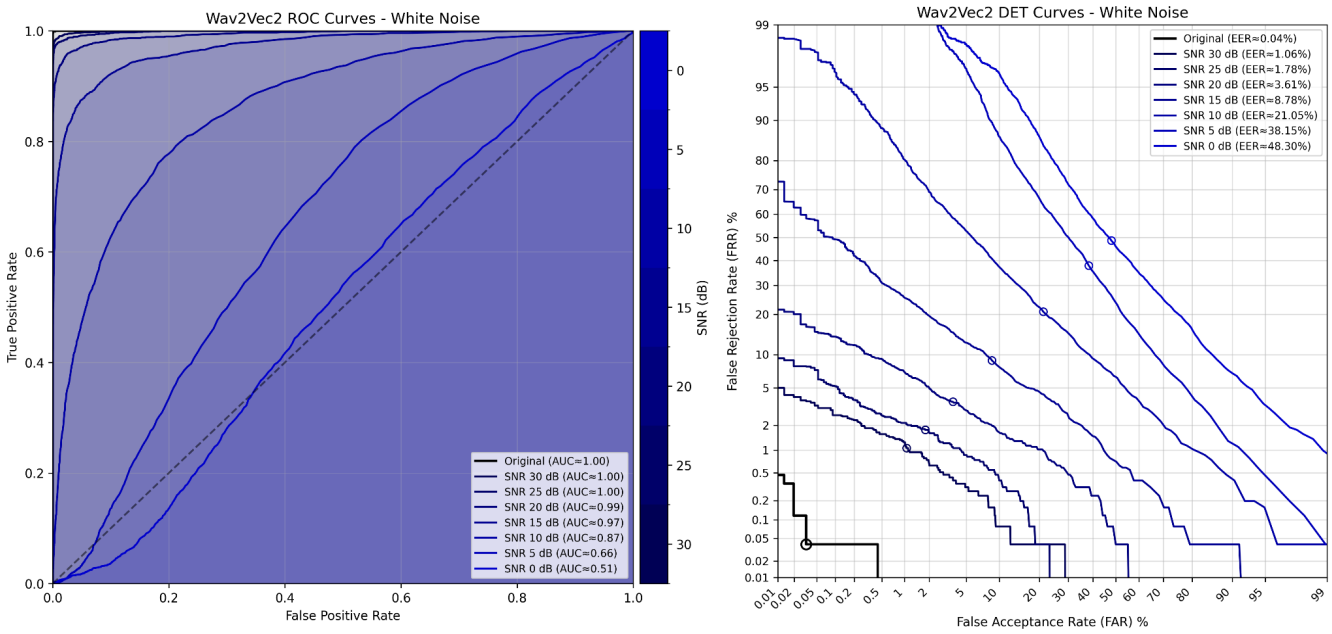

Figure 3: Wav2Vec2 ROC and DET Curves -White Noise

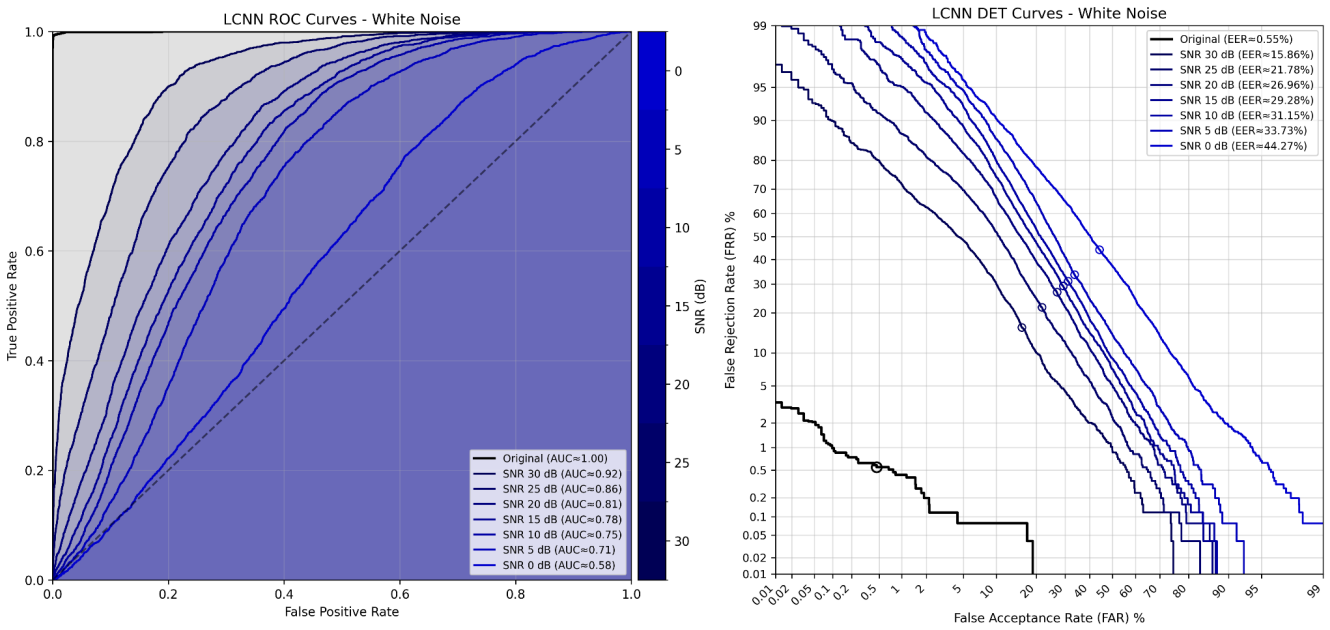

Figure 4: LCNN ROC and DET Curves -White Noise

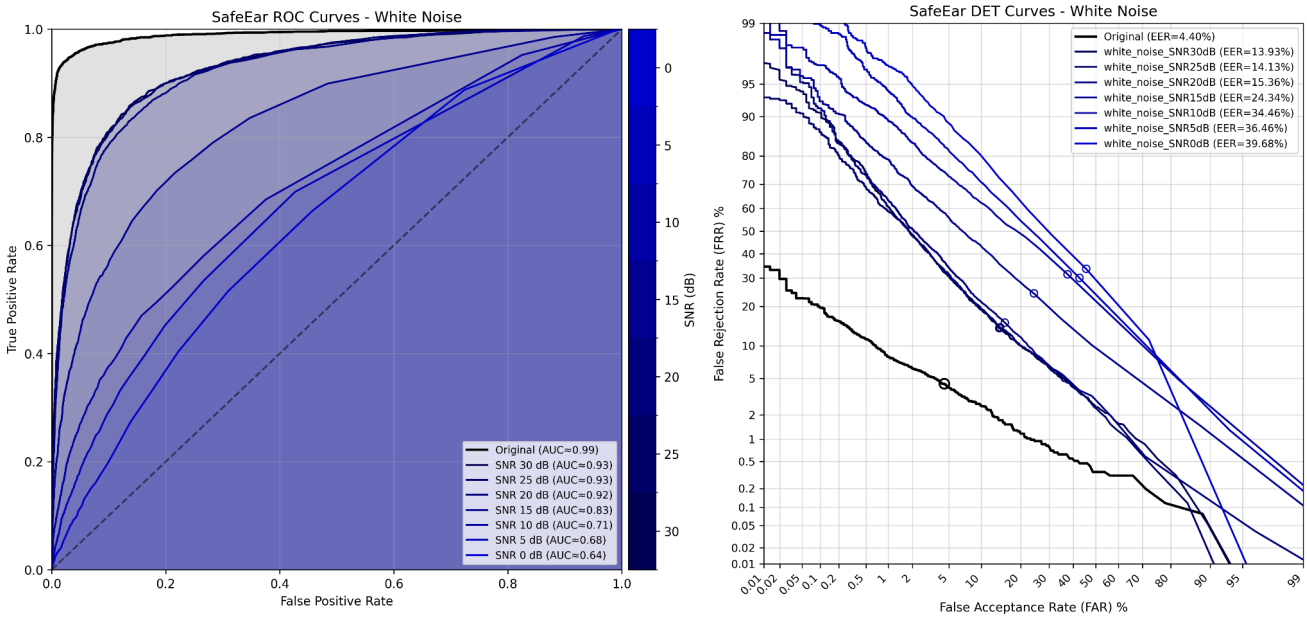

Figure 5: SafeEar ROC and DET Curves -White Noise

White noise was added to the test samples at 7 different signal-to-noise ratio (SNR) levels: 30 dB, 25 dB, 20 dB, 15 dB, 10 dB, 5 dB, and 0 dB. Figures 3, 4, and 5 show the ROC and DET curves under these conditions for Wav2Vec2, LCNN, and SafeEar, respectively. Across all models, performance degrades steadily as noise increases, or as SNR decreases in other words.

As seen in Figure 3, Wav2Vec2 has a near perfect performance on the original (clean) audio files, with an AUC at around 1.00 and EER of 0.04%. Its performance remains strong under mild noise: from 30 dB SNR to 20 dB SNR, the ROC curves stay in the top-left corner, and AUC remains above 0.99. Noticeable degradation begins at 15 dB SNR (AUC ≈ 0.97, EER ≈ 8.78%), indicating the model is starting to struggle with more corrupted inputs. The impact of white noise level becomes much more significant below 10 dB SNR, where the AUC value now is at 0.87 and the EER value at 21.05%. The most significant performance drops for the Wav2Vec2 model happens at 5 dB SNR (AUC ≈ 0.67, EER ≈ 38.15%) and 0 dB SNR (AUC ≈ 0.51, EER ≈ 48.30%), where classification is approaching random guessing. This trend is also reflected in the DET curves. As SNR decreases, the curves shift steadily upward and to the right, showing increases in both false rejection and false acceptance rates. The EER points, marked on the DET plot, clearly move away from the bottom-left corner as noise increases. While Wav2Vec2 remains highly robust down to 20 dB SNR its performance degrades rapidly at 15 dB and below, especially in high-noise environments.

LCNN in figure 4 shows good performance on clean audio, with an AUC near 1.0 and an EER of 0.55%, but it is slightly weaker than Wav2Vec2 even in the absence of noise. Howeverm its performance drops quickly as white noise is added. At 30 dB SNR, its AUC falls to 0.91, already a noticeable decline compared to Wav2Vec2. At 25 dB SNR, the drop becomes more significant (AUC ≈ 0.86, EER ≈ 21.78%), and by 20 dB SNR, LCNN's AUC falls to 0.81, with an EER of 26.96%. The 20 dB SNR marks the point where noise begins to seriously affect the model. From 15 dB downward, performance remains poor: AUC values stay below 0.78, and EERs rise above 29%, peaking at 44.27% at 0 dB SNR where classification is no longer reliable. The DET curves also clearly reflect this rapid decline. As the white noise level increases, the curves shift upward and to the right at a faster rate than Wav2Vec2's. The EER points visibly move further away from the ideal bottom-left position, showing growing error across all thresholds. In summary, LCNN begins to show noticeable degradation by 25 dB SNR, and its accuracy drops sharply below 20 dB SNR. This makes it less robust in noisy environments, particularly when compared to the more resilient Wav2Vec2.

SafeEar from Figure 5 performs reasonably well on clean audio, with an AUC around 0.99 and an EER of 4.40%, though both are lower than those of Wav2Vec2 and LCNN. As white noise increases, its ROC and DET curves shift downward and upward, respectively, indicating growing classification errors. Performance begins to degrade noticeably at 30 dB SNR, where the AUC drops to 0.93 and the EER rises to 13.93%. At 25 dB SNR and 20 dB SNR, SafeEar's performance remains relatively stable (AUCs around 0.92 and EERs around 14%), showing better resilience than LCNN. However, more severe degradation begins at 15 dB SNR (AUC $\approx$ 0.83, EER = 24.34%) and continues sharply at 10 dB SNR(AUC $\approx$ 0.71, EER = 34.46%) and 5 dB (AUC $\approx$ 0.68, EER $\approx$ 36.46%). At 0 dB SNR, SafeEar reaches an AUC of 0.64 and an EER of 39.68%, which, while high, is still better than both Wav2Vec2 (AUC $\approx$ 0.51, EER $\approx$ 48.30%) and LCNN (AUC $\approx$ 0.58, EER $\approx$ 44.27%) under this extreme noise condition.

Overall, Wav2Vec2 is the most reliable model when white noise is at low to moderate noise levels (SNR $\geq$ 15 dB), where it maintains strong accuracy with minimal degradation. SafeEar, while generally less accurate, shows better resilience in extremely noisy conditions (SNR $\leq$ 5 dB), outperforming both Wav2Vec2 and LCNN at 0 dB SNR. LCNN is the most sensitive to white noise overall, with faster performance decline and higher error rates across nearly all SNR levels.

4.2.2 Pink Noise

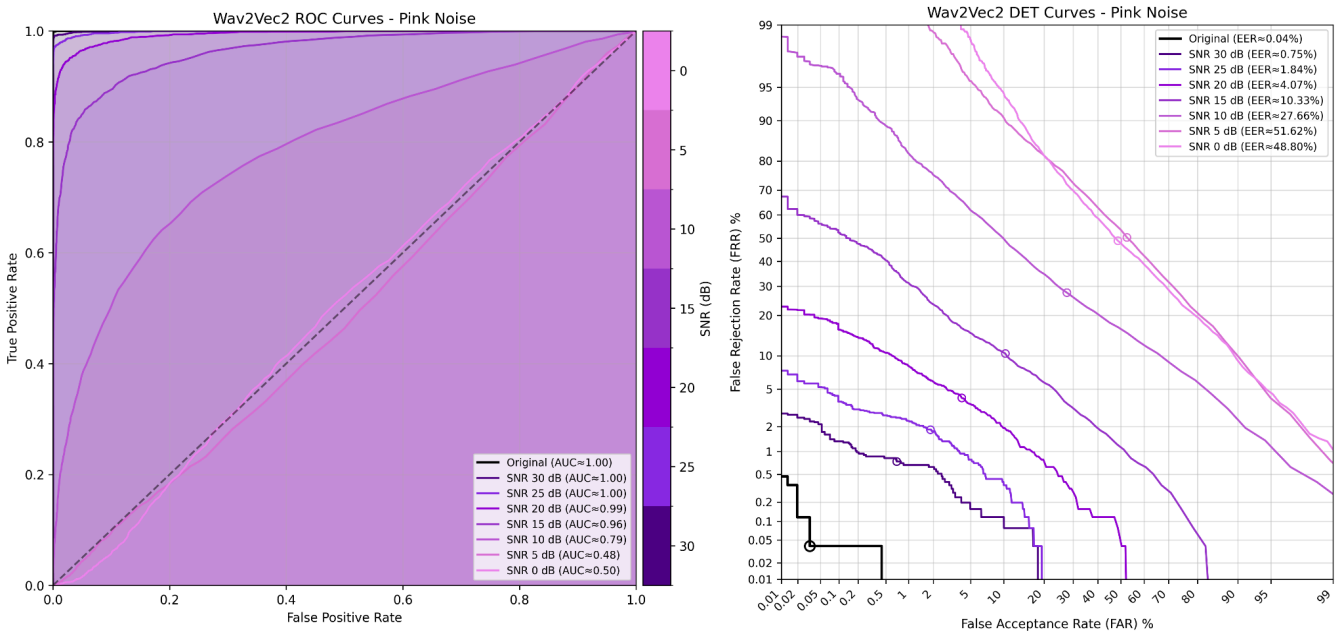

Figure 6: Wav2Vec2 ROC and DET Curves -Pink Noise

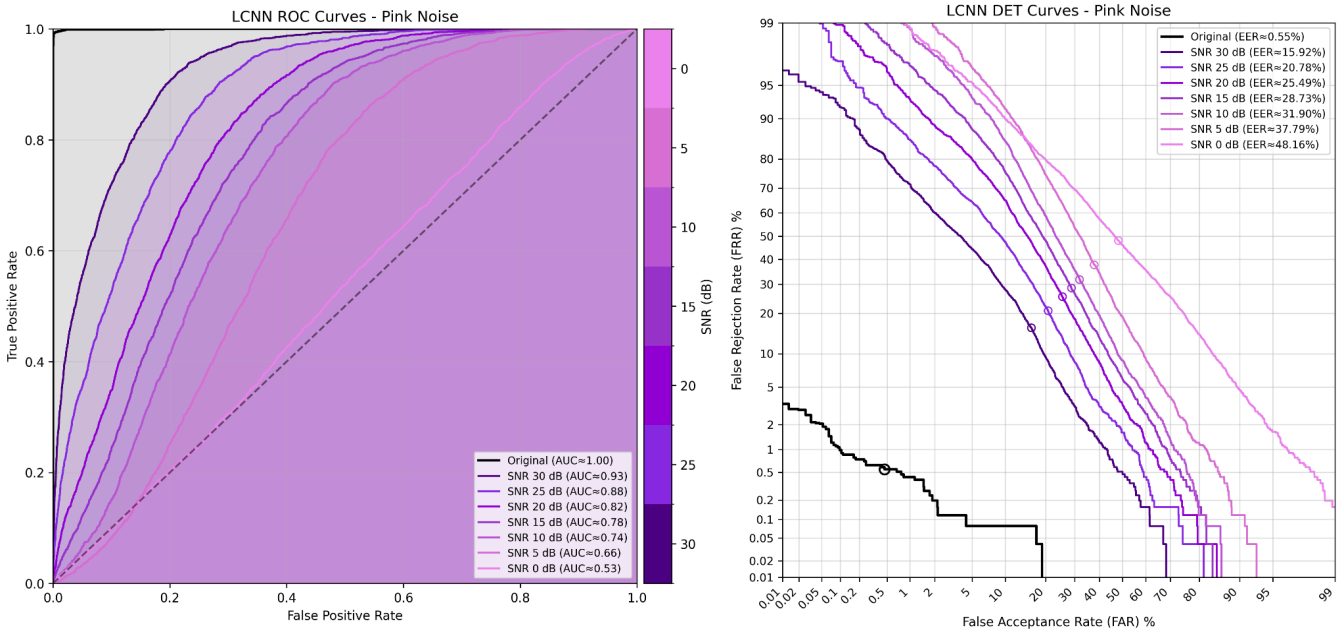

Figure 7: LCNN ROC and DET Curves -Pink Noise

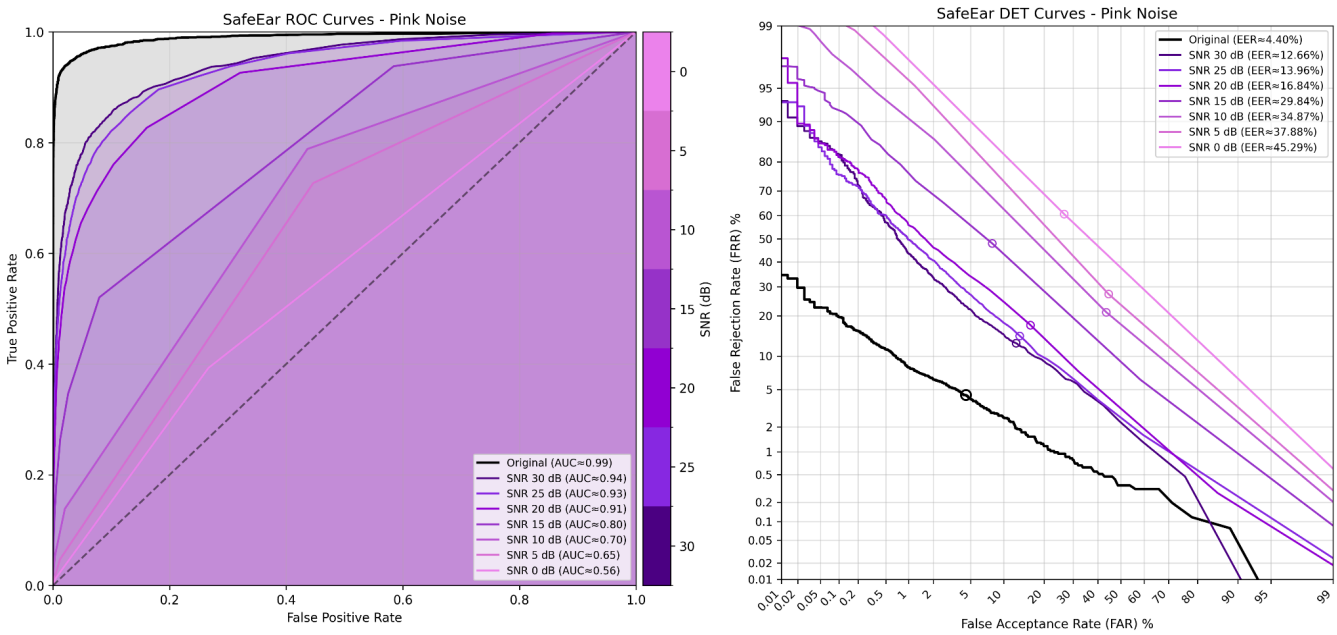

Figure 8: SafeEar ROC and DET Curves -Pink Noise

Pink noise was added in 7 levels as well to the original dataset. Looking at figure 6, Wav2Vec2 remains highly robust under pink noise at 30 dB SNR and 25 dB SNR, where AUC stays around 0.99 and EERs remain low (0.75% and 1.84%, respectively). The first signs of noticeable degradation appear at 20 dB SNR, where EER increases to 4.07%, though performance is still solid. At 15 dB SNR, the impact becomes more visible (AUC ≈ 0.96, EER ≈ 10.33%), and sharp declines follow continues. At 10 dB SNR, the AUC drop to 0.79 with EER rising to 27.66%, while 5 dB SNR results in a dramatic collapse (AUC ≈ 0.48, EER ≈ 51.62%), indicating worse-than-random predictions. The DET curves confirm this trend, with curves staying low and tight down to 20 dB SNR, then shifting upward rapidly from 15 dB SNR and below. In summary, Wav2Vec2 remains highly robust until about 20 dB SNR, with meaningful degradation beginning at 15 dB SNR and steep performance losses at 10 dB SNR and lower.

LCNN in figure 7 starts to degrade faster than Wav2Vec2 as pink noise level increases. At 30 dB SNR, AUC falls sharply from 1.00 to 0.93 and EER jumps from 0.55% to 15.92%, showing a clear loss of accuracy even with mild noise. The decline continues at 25 dB SNR (AUC = 0.8772, EER = 20.78%) and 20 dB SNR (AUC = 0.8217, EER = 25.49%), where the model starts showing serious errors. Below 15 dB SNR, LCNN's performance deteriorates further: AUC drops to below 0.78 and EER rises to

31.90%. At 5dB and 0 dB SNR, the predictions approach random guessing. Overall, both the ROC curves and the DET curves reflect a steep and steady shift as SNR decreases, indicating worsening false acceptance and rejection rates. Compared to Wav2Vec2, LCNN degrades more rapidly, especially below 25 dB SNR and is less able to recover in high-noise conditions.

For SafeEar, its degradation is more gradual at first in Figure 8. At 30 dB SNR, AUC drops slightly to 0.94 and EER rises to 12.66%. This pattern holds through 25 dB and 20 dB SNR with AUC values above 0.90 and EERs ranging from 13.96% to 16.84%, showing reasonable resilience to moderate noise. Its performance begins to degrade more noticeably at 15 dB SNR, where AUC falls to 0.80 and EER increases to 29.84%. From there, it worsens sharply: at 10 dB SNR, AUC drops to around 0.70 and EER reaches 34.87%, and by 0 dB SNR, SafeEar records an AUC of 0.56 and an EER of 45.29%, indicating highly unstable behavior under extreme noise. Its DET curves also show a consistent upward shift, but with a smoother gradient than LCNN, especially in the mid-SNR range.

In summary, pink noise augmentation affects all three models. Their ROC curves all shift consistently to the bottom right closer to the diagonal line as the noise level increases.Wav2Vec2 shows the strongest overall performance across most SNR levels. It maintains high accuracy down to 20 dB, with only moderate degradation at 15 dB, but its performance drops rapidly in more extreme noise conditions. LCNN is the most sensitive model, showing noticeable degradation as early as 30 dB, and continuing to decline steadily across the entire SNR range, becoming unreliable below 15 dB. SafeEar, while slightly behind Wav2Vec2 at higher SNRs, performs better than LCNN from 30 to 15 dB, and ultimately holds up best under very low SNRs (≤10 dB). Overall, Wav2Vec2 is the most effective under moderate noise, SafeEar proves more robust under extreme conditions, and LCNN degrades the most consistently and rapidly across the pink noise spectrum.

4.2.3 Brown Noise

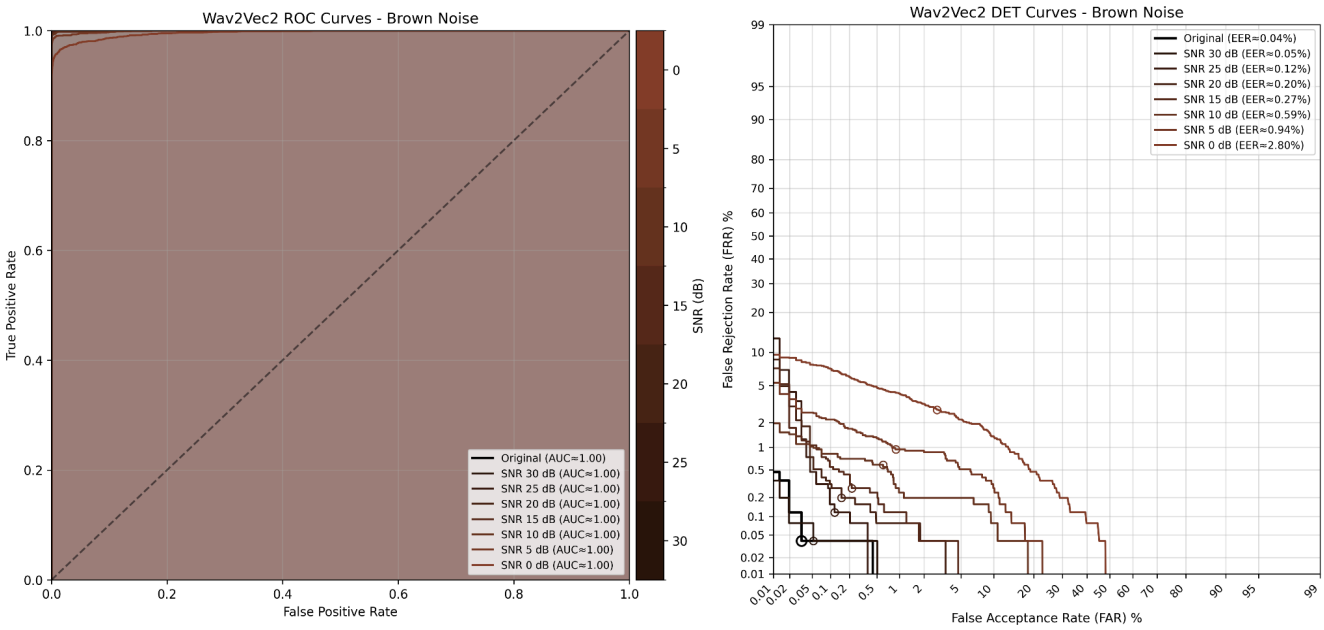

Figure 9: Wav2Vec2 ROC and DET Curves -Brown Noise

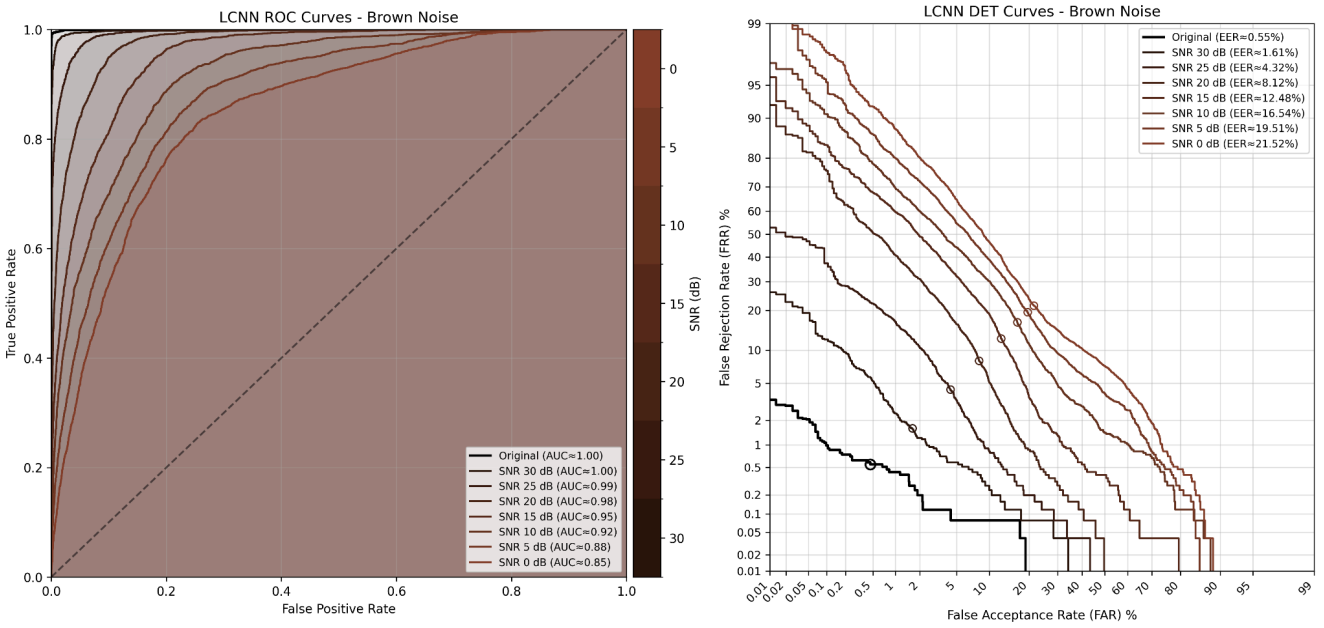

Figure 10: LCNN ROC and DET Curves -Brown Noise

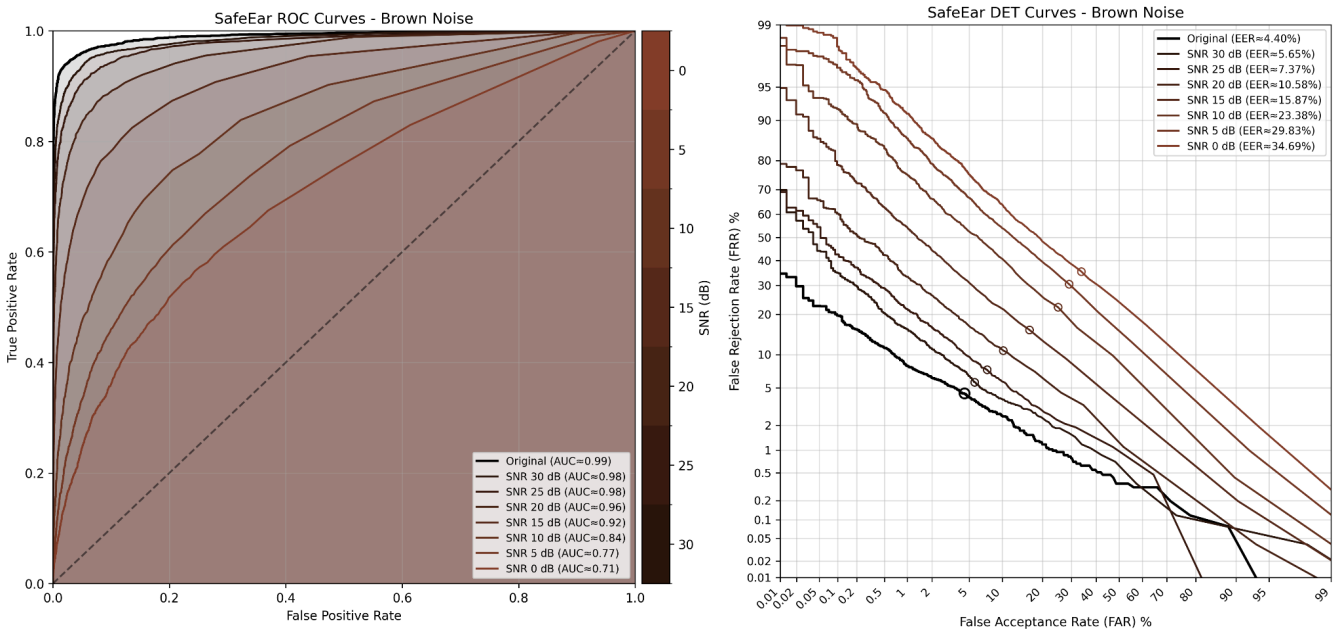

Figure 11: SafeEar ROC and DET Curves -Brown Noise

Under brown noise conditions, Wav2Vec2 demonstrates outstanding robustness, remaining near-perfect across all tested SNR levels. Looking at Figure 9, even at 0 dB SNR, Wav2Vec2 maintains an AUC near 1.00 and EER of just 2.80%. The ROC curves stay tightly clustered at the top-left corner across all SNR levels, and AUC values remain around 1.00. The DET curves show only a minor upward drift as noise increases, with EER rising gradually from 0.05% at 30 dB SNR to 0.94% at 5 dB SNR, before a slight jump to 2.80% at 0 dB SNR. In short, Wav2Vec2 is remarkably resilient to brown noise, showing minimal degradation

Looking at Figure 10, LCNN is noticeably weaker than Wav2Vec2 under brown noise. At 30 dB SNR and 25 dB SNR, the model remains stable (AUCs ≈ 1.00; EERs ≈ 1.61%, 4.32%), showing little impact from mild noise. However, performance starts to noticeably degrade at 15 dB SNR where AUC falls to 0.95 and EER rises to 12.48%, indicating that LCNN begins to struggle. This decline continues steadily through 10 dB (AUC ≈ 0.92, EER ≈ 16.54%), and becomes more severe by 5 dB and 0 dB SNR where AUCs drop to 0.88 and 0.85, and EERs rise above 19%. The DET curves clearly reflect this trend, with increasing errors at both low and high decision thresholds as noise intensifies.

In Figure 11, SafeEar also shows a steady, gradual degradation as brown noise increases, similar to LCNN. At 30 dB and 25 dB SNR, it maintains reasonably good performance, with both AUCs of 0.98, and EERs of 5.65% and 7.37%, respectively. As the brown noise level increases further, degradation becomes more noticeable: At 15 dB SNR, its AUC has fallen to 0.92 and EER reaches 15.87%. At 10 dB SNR the decline continues (AUC ≈ 0.84, EER ≈ 23.38%), and the model begins to struggle more noticeably. By 5 dB and 0 dB SNR, the performance deteriorates further with AUCs of 0.77 and 0.71, and high EERs of 29.83% and 34.69%, respectively. The ROC curves reflect this trend with a consistent downward shift, and DET curves show widening error as both false acceptance and rejection rates increase.

Overall, Wav2Vec2 is the most robust, maintaining high accuracy across all SNR levels with minimal degradation under brown noise, even at 0 dB SNR. LCNN performs reasonably well up to 20 dB SNR but degrades steadily at lower SNRs, though it remains usable at 0 dB SNR. SafeEar shows smooth but weaker performance, consistently ranking below both models at every SNR level. In short, Wav2Vec2 leads, followed by LCNN, with SafeEar as the least effective under brown noise.

4.2.4 Compare Across Three Noise Types

Across all three noise types, Wav2Vec2 is the most robust overall, performing best under brown noise with minimal degradation even at 0 dB SNR. It handles pink noise well down to 20 dB SNR but drops sharply below that, and performs the worst under white noise, with early and severe degradation. LCNN is most stable under brown noise, with gradual decline, but is highly sensitive to white and pink noise, showing significant drops even at high SNRs. SafeEar is less accurate overall but more resilient in extreme white and pink noise (≤ 5 dB SNR), where it outperforms the other two. However, it consistently ranks lowest under brown noise. Overall, Wav2Vec2 leads in general noise robustness, LCNN is second under brown noise, and SafeEar is most stable in high-noise extremes for white and pink noise.

4.3 Codec-based Augmentation Performance

Having compared the models' robustness across different noise types, the next focus shifts to codec-based augmentations, which simulate the effects of audio

compression commonly encountered in telecommunication and streaming platforms. Six codec formats are selected: ulaw, opus, g722, gsm, alaw, and mp3.

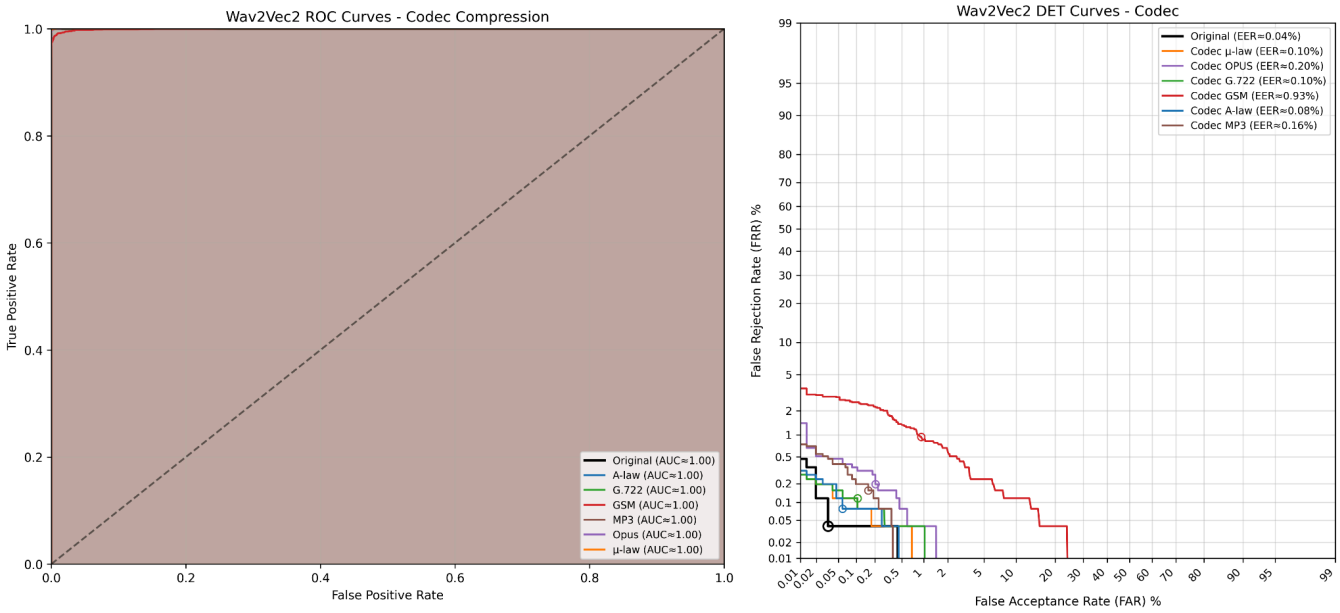

Figure 12: Wav2Vec2 ROC and DET Curves -Codec

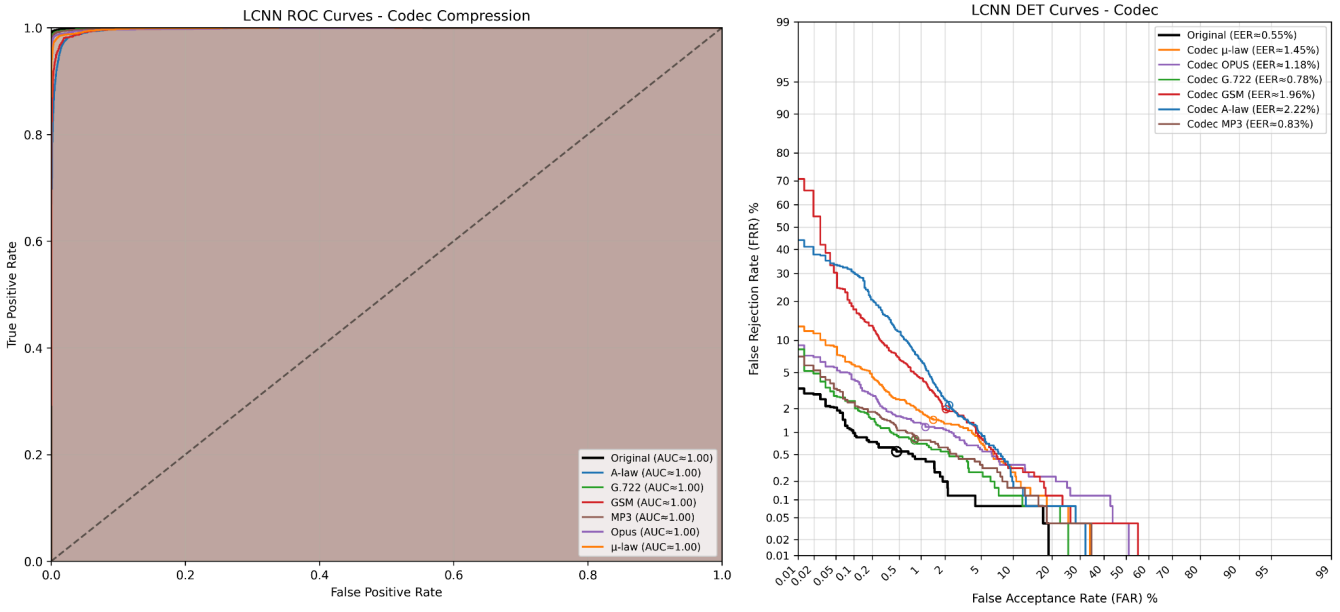

Figure 13: LCNN ROC and DET Curves -Codec

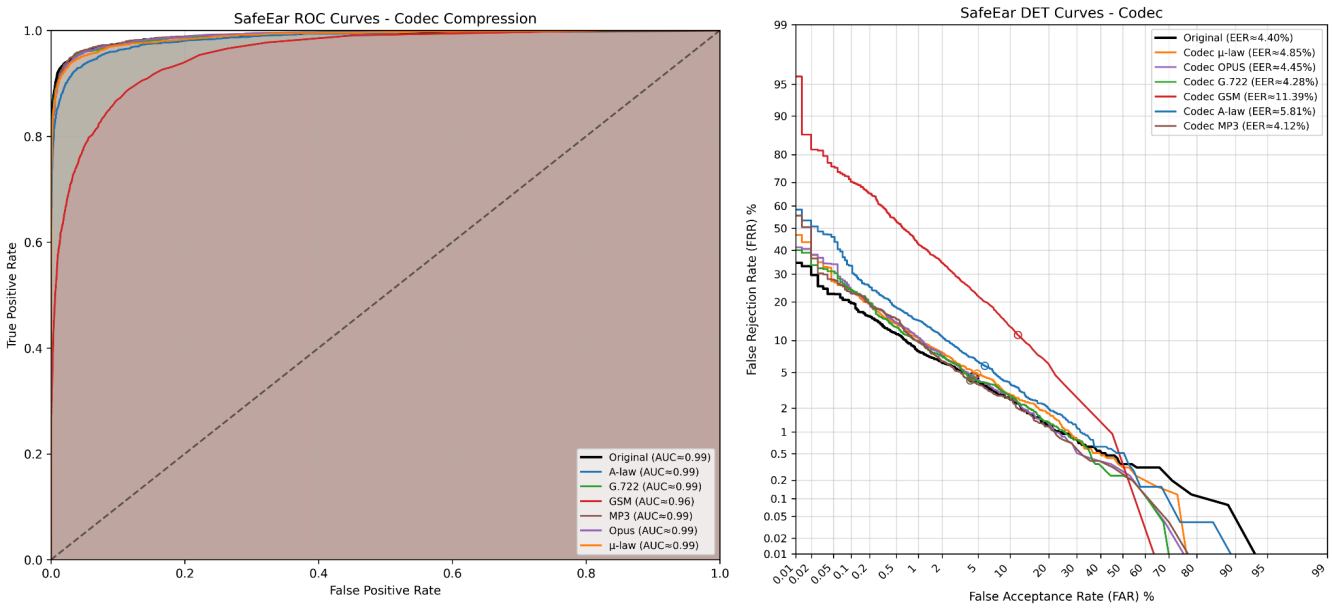

Figure 14: SafeEar ROC and DET Curves -Codec

From figure 12 to 14, the ROC and DET curves for codec-based augmentations show that all three models—Wav2Vec2, LCNN, and SafeEar—maintain performance close to their original (clean) baselines across various codecs. In Figure 12, the curves for Wav2Vec2 remain tightly clustered near the top-left of the ROC plot and bottom-left of the DET plot, with only a slight shift for the GSM codec. In Figure 13, LCNN shows a similarly minor spread among curves, with GSM and ALAW codecs introducing slightly higher error. SafeEar's DET curves from Figure 14 also follow a consistent pattern, with GSM showing the most deviation, but overall variation is small. Across models, the codec-induced distortions do not cause major separations in the curves, and AUC values remain high while EERs stay low.

Therefore, the plots suggest that codec compression has minimal impact on all three models' performances. Wav2Vec2 is the best performing model among all codecs, followed closely by LCNN and SafeEar. Overall, codec-based distortions are far less harmful than noise-based ones.

4.4 Time-stretch Augmentation Performance

While codec-based compression had minimal impact on model performance, time-stretch augmentations introduce a different type of challenge by altering the temporal characteristics of the audio signal. In this section, time-stretch augmentation is

separated into two segments: stretch slower and stretch faster. Each segment follows a consistent step size of 10% (or 0.1 in tempo factor). The stretch slower segment includes nine tempo factors ranging from 0.9× down to 0.1×, while the stretch faster segment includes ten conditions from 1.1× up to 2.0×. Below, the performance under the stretch slower conditions is described with reference to the ROC and DET plots.

4.4.1 Stretch Slower

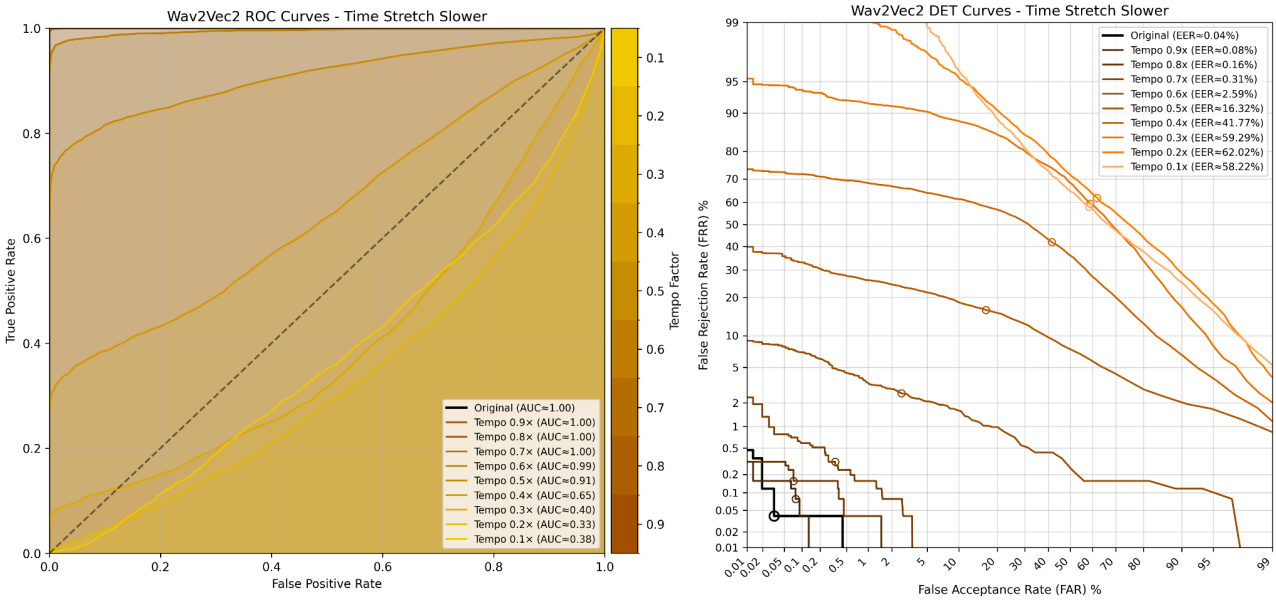

Figure 15: Wav2Vec2 ROC and DET Curves -Time Stretch Slower

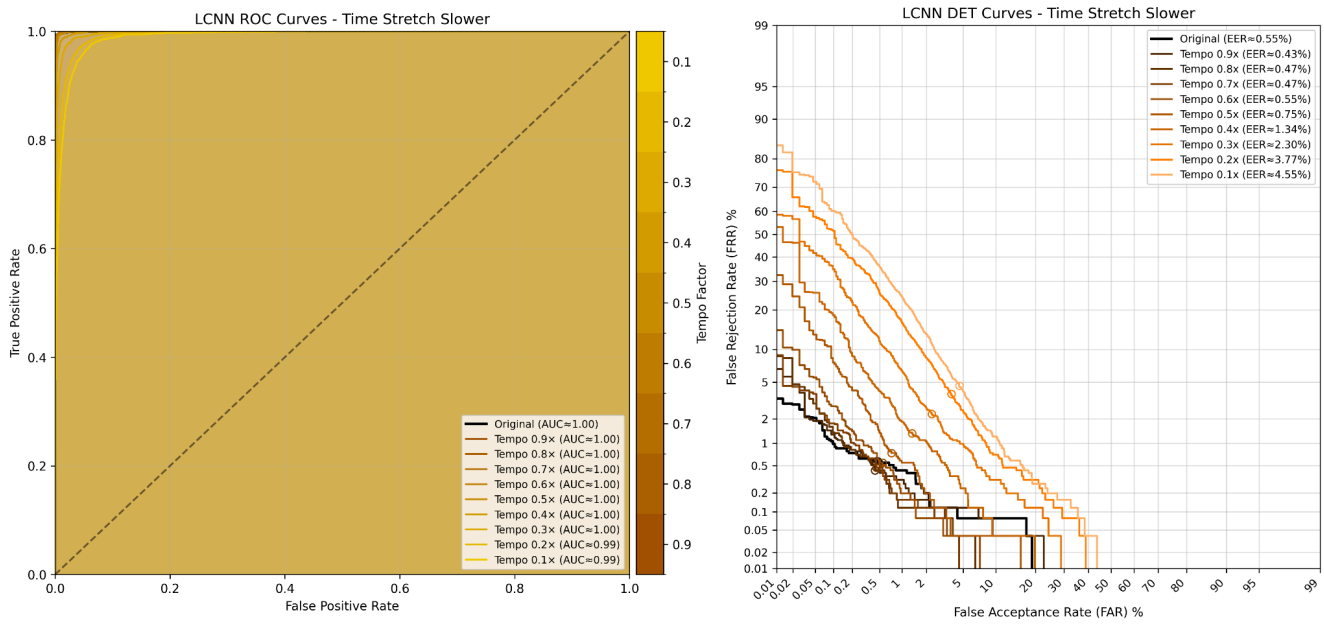

Figure 16: LCNN ROC and DET Curves -Time Stretch Slower

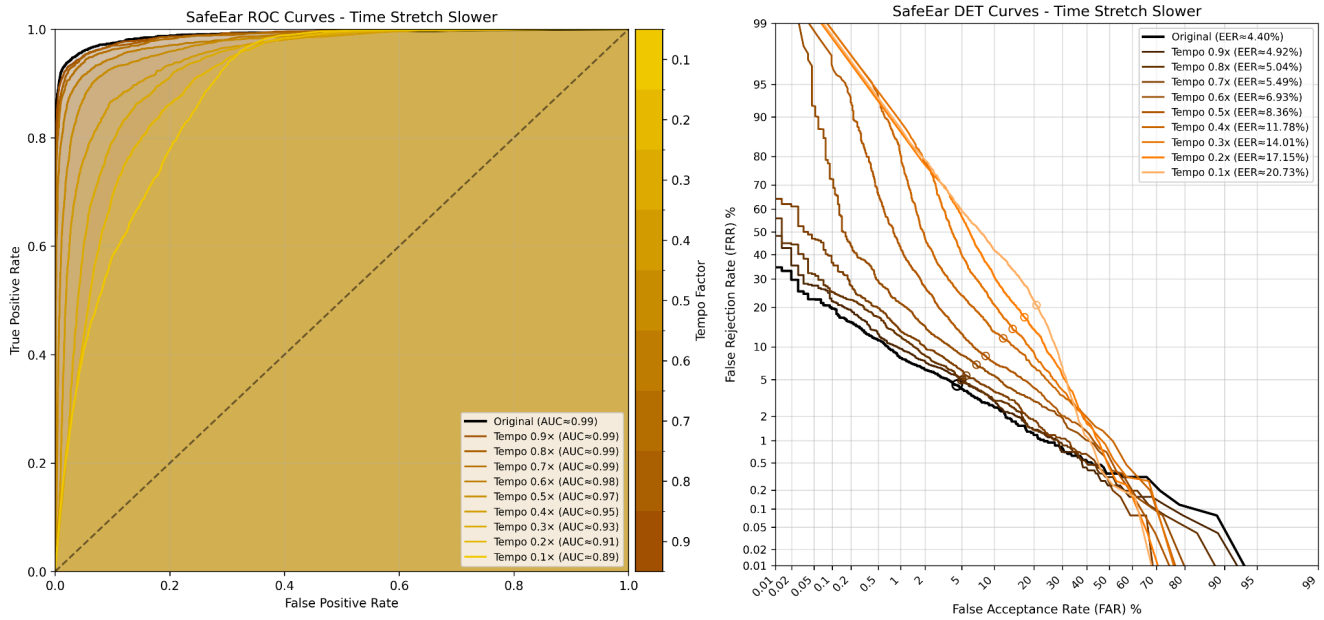

Figure 17: SafeEar ROC and DET Curves -Time Stretch Slower

Wav2Vec2 in Figure 15 shows strong robustness to mild slowdowns. At tempo factors 0.9× to 0.7×, the ROC curves stay close to the top-left corner, and DET curves remain low, with AUC at about 1.00 and EERs under 0.31%, indicating minimal impact. A slight drop begins at 0.6× (AUC = 0.99, EER = 2.59%). From there, performance

degrades significantly. At 0.5×, the ROC curve shifts to the bottom largely. Its AUC also drops to 0.91 and EER rises to 16.32%. At 0.4×, the ROC curve flattens further (AUC = 0.65), and DET error rises sharply (EER = 41.77%). After been slowed down to 0.3× speed and below, the ROC curves are even below the diagonal line and AUCs are below 0.5, showing Wav2Vec2 is predicting randomly.

LCNN, on the other hand, handles slow tempo changes very well. As presented in Figure 16, across all stretch conditions (from 0.9× down to 0.1×), the ROC curves remain tightly grouped near the top-left corner, with AUC values consistently at or above 0.99, and often 1.00. The DET curves also show minimal upward shift, with EERs ranging modestly from 0.43% to 1.34%. The DET curves also show very little shift, with EERs gradually rising from 0.43% (0.9×) to just 4.55% at the most extreme 0.1× speed. Even at high levels of distortion, LCNN maintains a stable performance profile.

In Figure 17, SafeEar shows moderate sensitivity to slower tempo changes. The ROC curves shift gradually downward, with AUC decreasing from 0.99 (original and 0.9×) to 0.89 at 0.1×. The DET curves also show a consistent upward trend, with EER increasing from 4.40% to 20.73%. Unlike Wav2Vec2, which experiences sharp breakdowns at specific points, SafeEar degrades steadily but not catastrophically, preserving some detection capability even at slower tempos.

Overall, under slow time-stretching, all three models maintain strong performance at mild slowdowns (tempo $\geq 0.6\times$), but differences emerge under heavier distortion. Wav2Vec2 is the most sensitive to slowdowns, with AUC falling below 0.5 and EER rising sharply to over 58% at 0.1×. LCNN is the most robust, maintaining excellent accuracy even at extreme slowdown levels, with only a modest increase in EER. SafeEar lies between the two, showing consistent but moderate degradation. In short, LCNN handles slow time-stretching the best, while Wav2Vec2 struggles the most under extreme temporal distortion.

4.4.2 Stretch Faster

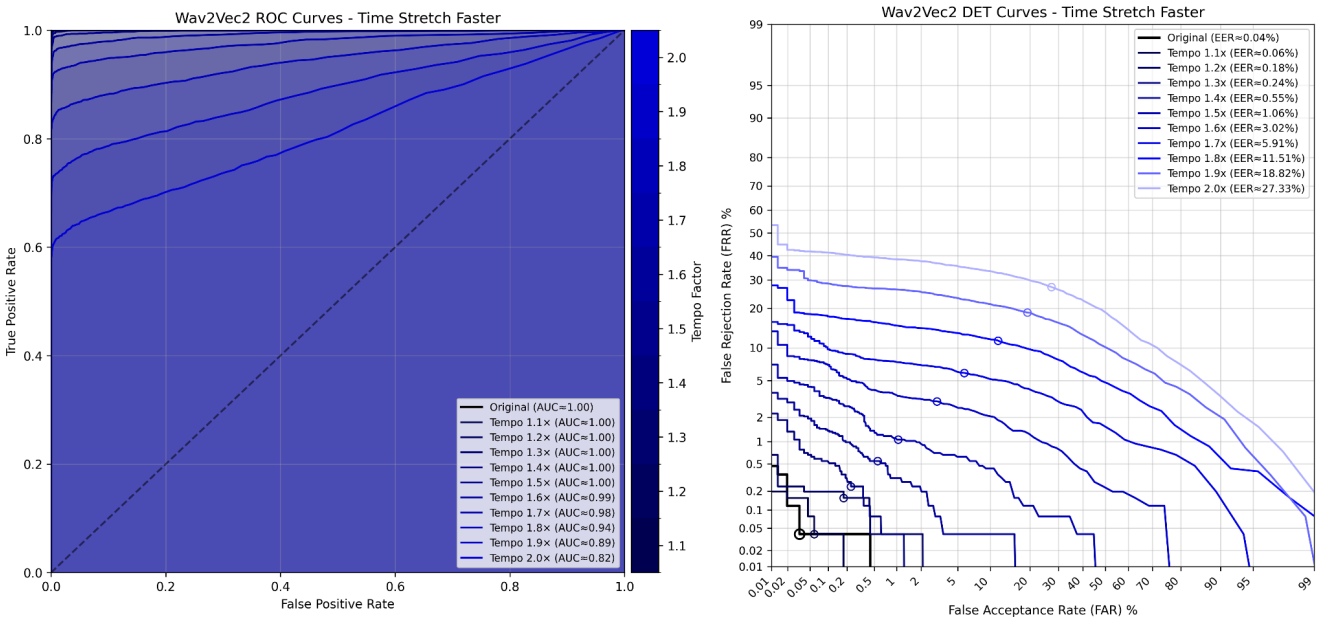

Figure 18 : Wav2Vec2 ROC and DET Curves -Time Stretch Faster

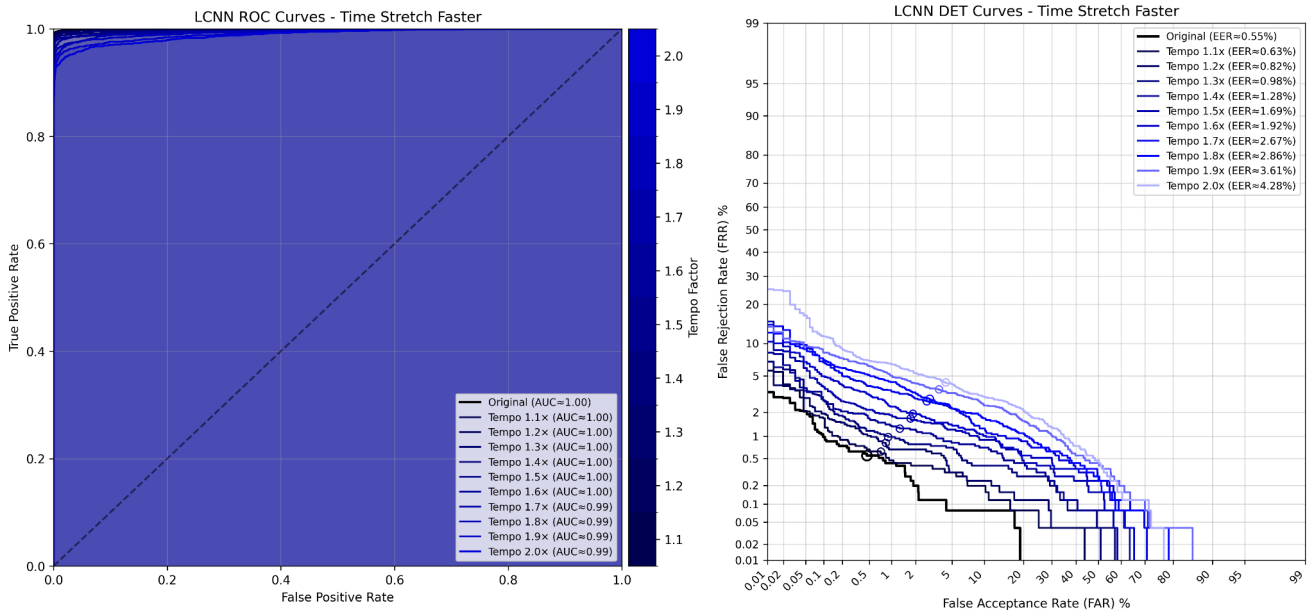

Figure 19: LCNN ROC and DET Curves -Time Stretch Faster

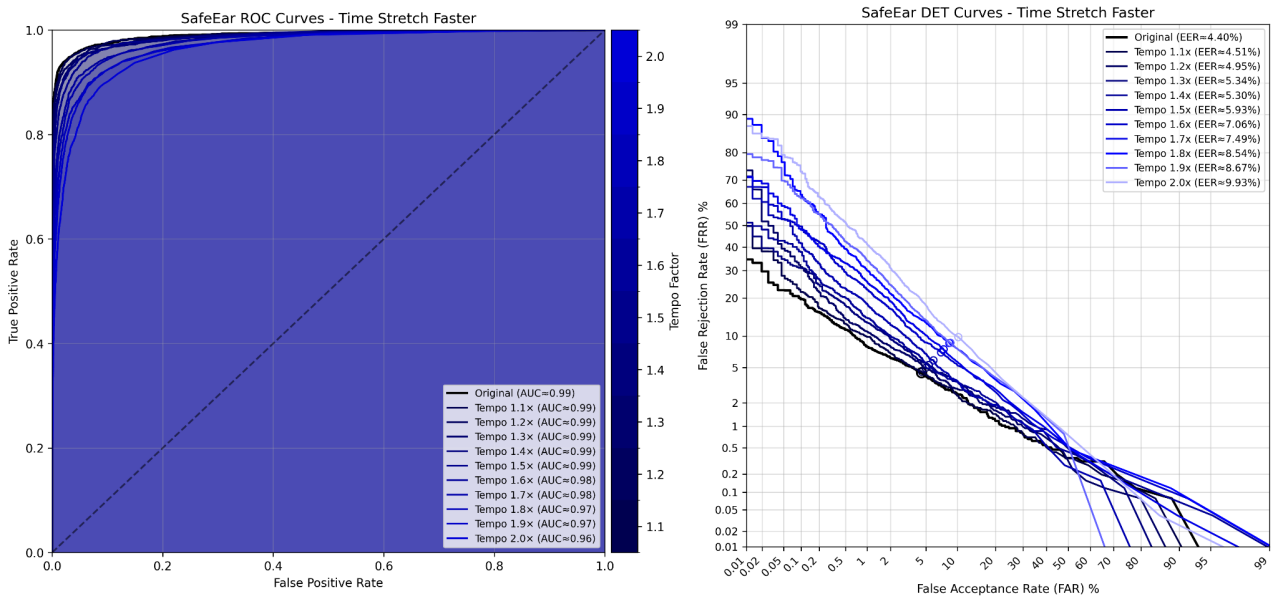

Figure 20: SafeEar ROC and DET Curves -Time Stretch Faster

Observing from Figure 18, Wav2Vec2 shows strong robustness under mild speed-ups as well. From 1.1× to 1.7×, the ROC curves remain close to the top-left corner, and AUC stays above 0.98, with low EERs ranging from 0.06% to 5.91%, indicating minimal impact. A notable performance drop begins at 1.8×, where the AUC falls to 0.94 and the EER jumps to 11.51%, marking the point where speeding up begins to noticeably impair the model. The decline continues at more extreme speed up (1.9× and 2.0× speed up), where the ROC curve flattens and the DET curve shifts far upward. During this stage, AUC drops into the 0.8 range while EER climbs sharply to 27.33%, indicating a more significant degradation in detection performance.

In Figure 19, LCNN remains very robust across all tempo acceleration levels. Its ROC curves remain tightly packed. From 1.1× to 2.0×, the AUC stays consistently at or near 1.00 while the DET curves only shift gradually, with EERs rising from 0.63% (1.1×) to 4.28% at 2.0×. From the results of the figure, LCNN has the lowest sensitivity to fast time-stretching among the three models, reflecting stable classification confidence across speeding up conditions.

SafeEar in Figure 20 demonstrates moderate robustness under increasing tempo. Its AUC remains stable at 0.99 from 1.1× to 1.5×, then decreases gradually since 1.6× speed up. At the most extreme speed up, its AUC is still at 0.96 with EER below 10%.

Compared to Wav2Vec2, SafeEar degrades more gently and maintains more consistent performance in higher tempo ranges.

4.4.3 Summary for Time-stretch Augmentation Performance

Across time-stretch augmentations, all models are generally more vulnerable to slow stretching than to fast stretching.Wav2Vec2 is the most sensitive to extreme slowdowns, showing sharp drops in AUC and steep rises in EER, especially below 0.5×. In contrast, it handles speeding up tempo well until performance begins to noticeably degrade at 1.8×. LCNN is the most robust overall, showing minimal change across all stretch factors. SafeEar degrades steadily under slowdowns but maintains stable performance under fast stretches. Overall, slow time-stretching poses a greater challenge than fast, and LCNN consistently shows the highest resilience among the three models.

4.5 Pitch-shift Augmentation Performance

Building on the findings from the time-stretch augmentation section, which highlighted the impact of temporal alterations on model performance, we now turn to pitch-shift augmentation. Pitch-shifting introduces frequency-based distortions by altering the pitch of the audio signal while preserving its duration. This section evaluates how upward and downward pitch shifts affect the performance of deepfake detection models. A total of twemty conditions are tested—10 pitch shift up (+1 to +10 semitones) and six pitch shift down (–10 to –10 semitones). These shifts simulate realistic variations such as tuning inconsistencies, playback alterations, or adversarial manipulations. The analysis below examines how each model handles these spectral changes.

4.5.1 Pitch Shift Up

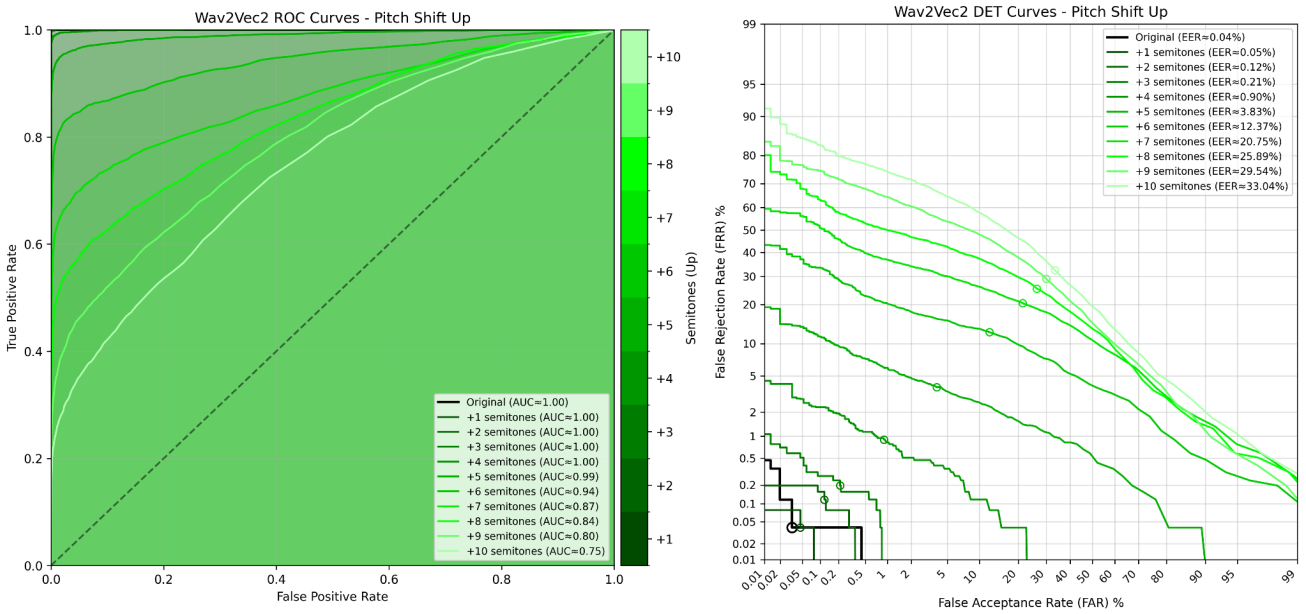

Figure 21: Wav2Vec2 ROC and DET Curves -Pitch Shift Up

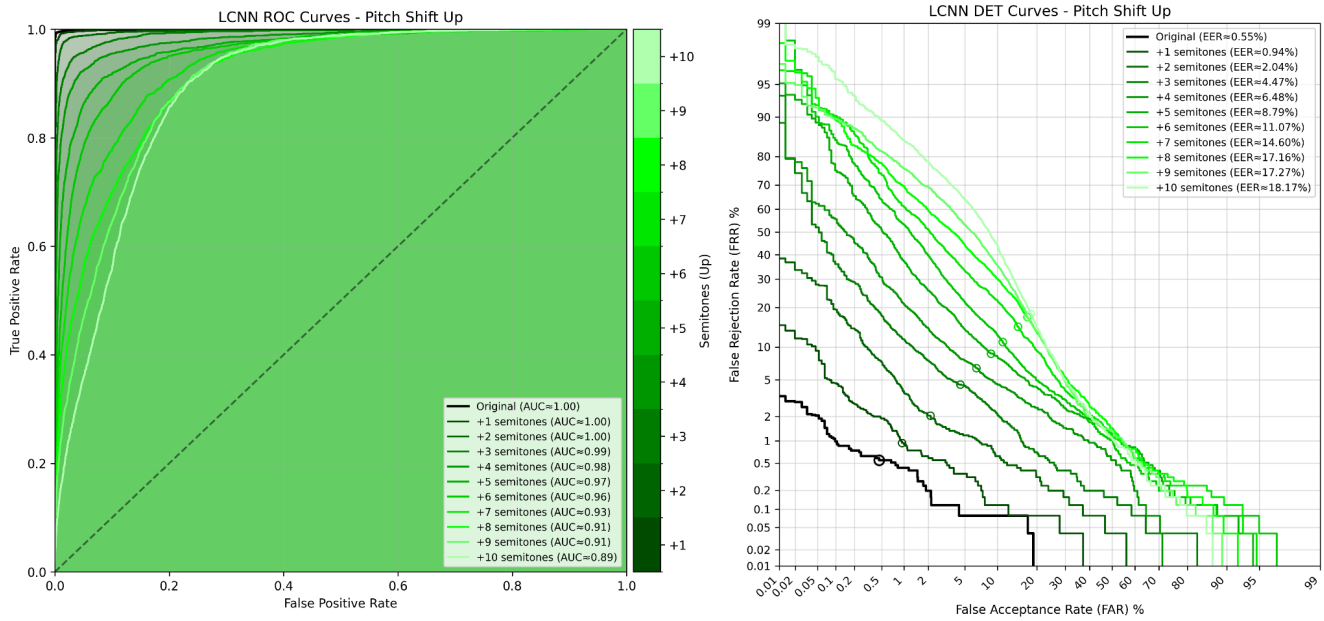

Figure 22: LCNN ROC and DET Curves -Pitch Shift Up

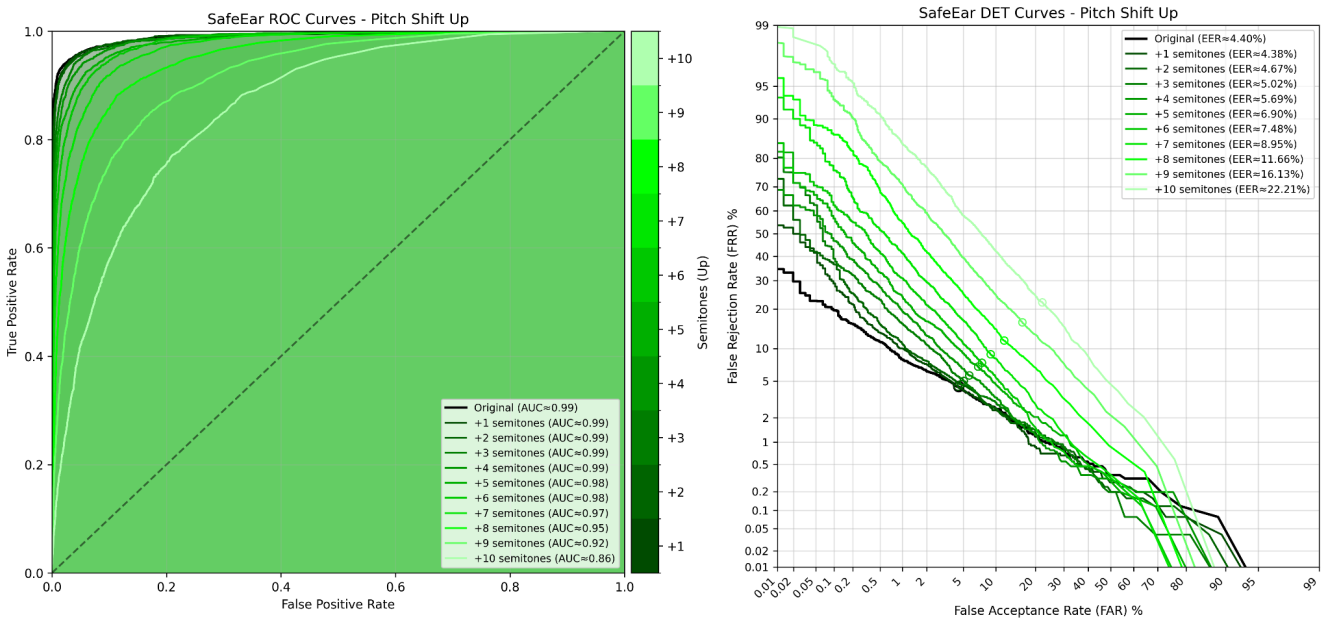

Figure 23: SafeEar ROC and DET Curves -Pitch Shift Up

Under pitch shift up conditions, Wav2Vec2 demonstrates strong resilience to moderate pitch increases. As shown in Figure 21, the ROC curves remain tightly clustered in the top-left region from +1 to +5 semitones, with AUC values consistently above 0.99 and EERs below 4%. At +6 semitones, performance starts to drop more noticeably (AUC = 0.94, EER = 12.37%). Beyond that, the trend continues sharply: +7 to +10 semitones bring substantial declines (AUC down to 0.75, EER up to 33.04% at +10). The DET curves also reflect this trend clearly, shifting steeply upward as pitch increases, indicating Wav2Vec2 becomes increasingly error-prone under larger pitch shifts.

LCNN, shown in Figure 22, experiences a more gradual but steady degradation. The ROC curves slowly shift downward with increasing pitch, with AUC falling from 1.00 to 0.89 by +10 semitones. EER rises from 0.55% (original) to 18.17% at the extreme, with more visible degradation occurring from +5 semitones onward. While LCNN maintains stronger performance than Wav2Vec2 at extreme pitch changes (e.g., +7 and above), its overall slope shows that it is still affected by frequency alterations to some degrees.

Looking at figure 23, SafeEar experiences moderate but steady degradation under upward pitch shifts. Its AUC drops from 0.99 (original) to 0.86 at +10 semitones, and EER increases from 4.40% to 22.21%. The ROC curves shift downward gradually, and

DET curves rise at a steady pace, reflecting consistent but less controlled performance decline compared to LCNN. SafeEar performs reasonably well under mild shifts (+1 to +4 semitones), but its error rate grows more rapidly beyond +6, same as LCNN and Wav2Vec2.

Overall, all three models show decreasing performance as pitch shift increases, but the rate and pattern of degradation differ. +6 semitones marks a key turning point for Wav2Vec2 and LCNN models, where performance noticeably worsens. SafeEar does not have sharp collapse, but it experience steady degradation. Wav2Vec2 is highly accurate at small shifts but degrades rapidly beyond +6, with sharp drops in AUC and rising EER. LCNN maintains the most consistent and reliable performance across the full range, with a gradual AUC decline and lower EERs even at higher shifts though with a little jump of EER at +6 semitones. SafeEar handles moderate shifts reasonably well but becomes less stable and more error-prone after +6, especially compared to LCNN. These results suggest that LCNN is the most robust under pitch-based distortions.

4.5.2 Pitch Shift Down

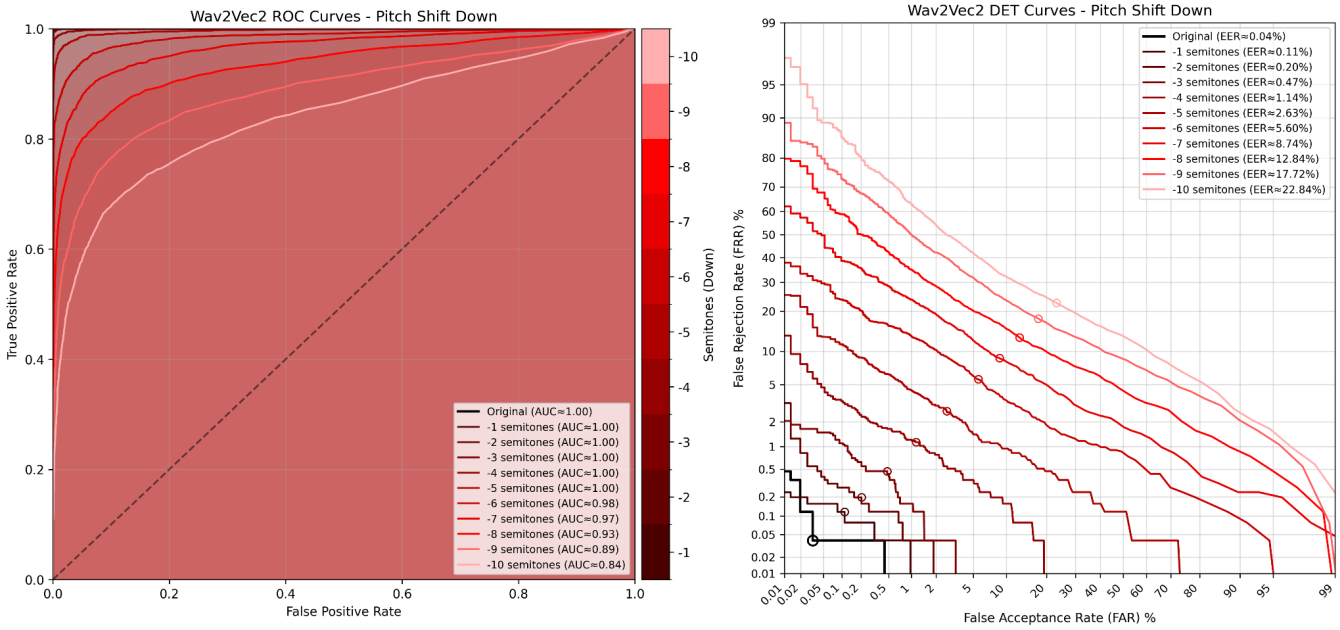

Figure 24: Wav2Vec2 ROC and DET Curves -Pitch Shift Down

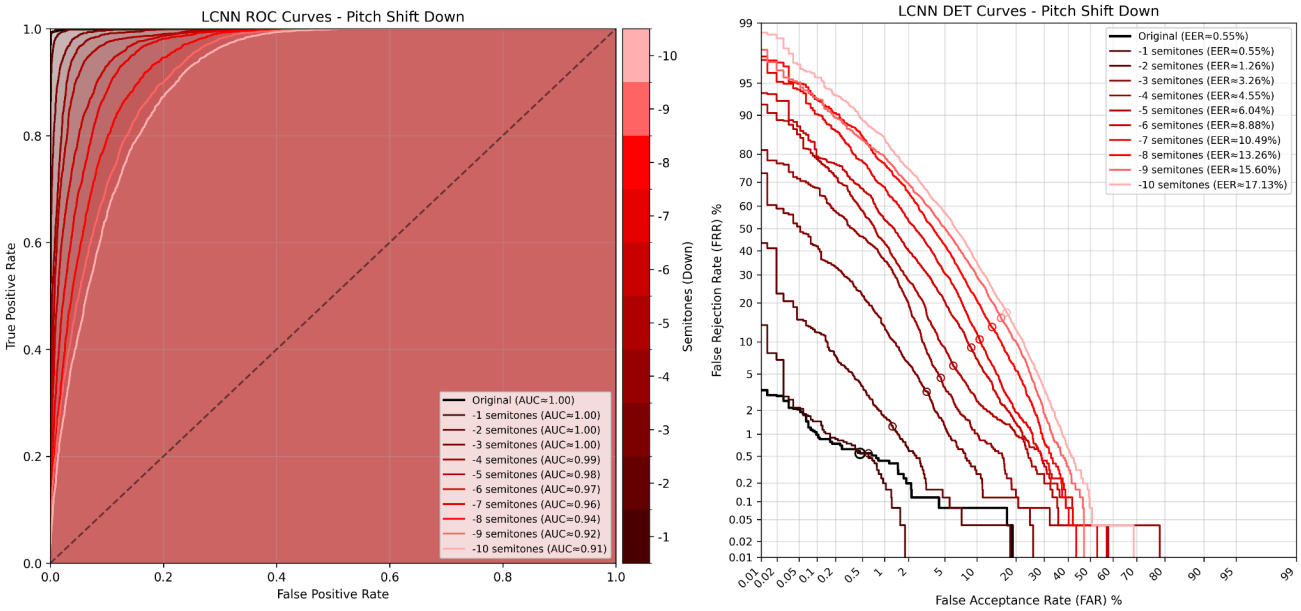

Figure 25: LCNN ROC and DET Curves -Pitch Shift Down

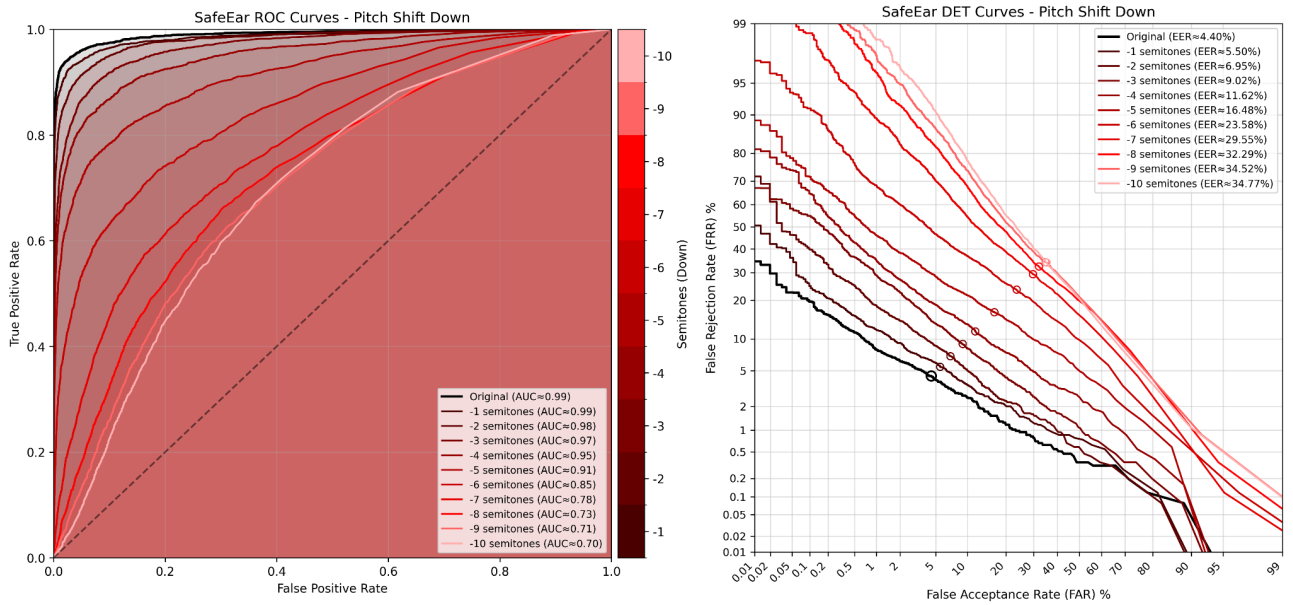

Figure 26: SafeEar ROC and DET Curves -Pitch Shift Down

Under downward pitch shift conditions, Wav2Vec2 performs very well up to moderate changes. As shown in Figure 24, its ROC curves stay tightly clustered near the top-left corner through –5 semitones, with AUC values holding at 1.00 and EERs below 2.63%. Performance degradation begins to emerge more clearly at –6 semitones (AUC = 0.98, EER = 5.60%) and becomes increasingly evident from –7 to –10 semitones, where

AUC declines steadily to 0.84 and EER rises to 22.84%. The DET curves reflect this trend, shifting upward with each increment, indicating growing classification difficulty under lower-pitched inputs. While Wav2Vec2 handles moderate downward shifts robustly, significant degradation starts to set in just past the –6 semitone point.

LCNN, shown in Figure 25, also begins with high performance (AUC = 1.00, EER = 0.55%) and retains strong results under small pitch changes. From –1 to –4 semitones, AUC remains above 0.99, and EER stays below 5%. Degradation becomes more noticeable beyond –5 semitones, with the AUC dropping to 0.97 at –6 and continuing down to 0.91 by –10. EER similarly increases to 17.13% at –10. Although performance does decline, LCNN exhibits slower and more controlled degradation compared to Wav2Vec2, and retains a usable detection margin even under substantial downward shifts.

SafeEar, shown in Figure 26, demonstrates more severe sensitivity to pitch drops. Starting from a lower baseline (AUC = 0.99, EER = 4.40%), its performance begins declining more steeply than the other models. From –3 semitones onward, ROC curves flatten and DET curves shift significantly upward. At –6 semitones, AUC falls to 0.85, and EER spikes to 23.58%. This decline continues sharply through –10, where AUC reaches 0.70 and EER climbs to 34.77%. Therefore, SafeEar has the weakest performance among the three models at these levels. These results indicate that SafeEar is highly vulnerable to downward pitch alterations.

In summary, all three models exhibit decreasing performance with increasing downward pitch shifts. Wav2Vec2 handles moderate shifts (up to –5 semitones) well, but its performance drops significantly beyond –6. LCNN shows the most consistent degradation curve, maintaining relatively strong performance even at the highest shifts. SafeEar degrades most rapidly and struggles to retain accuracy under extreme pitch decreases.

4.5.3 Summary for Pitch-shift Augmentation Performance

Among the three, LCNN again proves to be the most robust under frequency-domain distortions, while Wav2Vec2 performs reliably up to a critical threshold, and SafeEar is the most affected by pitch shifting downward. Across both pitch

shift directions, all three models show stable performance under mild conditions, but ±6 semitones consistently mark the point at which noticeable performance degradation begins.

Wav2Vec2 maintains high accuracy and low EER up to ±4 semitones, but its performance drops sharply beyond ±6, with AUC and DET curves indicating increasing misclassification. LCNN handles pitch changes most consistently, with a gradual and controlled decline in performance even under extreme shifts. SafeEar degrades more steadily under pitch increase, but its performance deteriorates rapidly when pitch is shifted downward, showing the steepest rise in EER and the lowest AUC under –6 semitones and beyond.

In short, a shift of ±6 semitones marks a common threshold where model robustness begins to noticeably deteriorate. This point distinguishes reliable performance under mild distortions from clear vulnerabilities under more significant spectral changes. Among the models, LCNN demonstrates the highest resilience, while Wav2Vec2 and SafeEar show greater susceptibility beyond this critical boundary.

4.6 Reverb Augmentation Performance

This section examines how reverberation affects model performance. Reverb, typically produced in reflective environments like large rooms or corridors, introduces temporal smearing in audio signals, which can mask important speech features. To evaluate model robustness, real-world impulse responses (IRs) were used for convolution, enabling the simulation of authentic acoustic characteristics from various environments. The IRs were categorized into three types: small rooms (e.g., bathroom, car interior, elevator), large rooms (e.g., church, train station, Elveden Hall), and open spaces (e.g., beach, forest, park). These categories represent a spectrum of reverberation times (RT30) and spatial characteristics, allowing for a comprehensive and realistic assessment of model performance across diverse acoustic scenarios.

4.6.1 Small Room

The small room category includes IRs from highly enclosed environments with short reverberation times, generally ranging from 0.04 s to 0.34 s. These spaces produce quick-decaying reflections, which typically do not interfere heavily with speech clarity.

Among the three IRs in this group, the elevator interior is the most reverberant (0.34 s), followed by the bathroom (0.26 s) and the car interior, which has the shortest decay time at only 0.04 s. These conditions simulate common compact indoor settings.

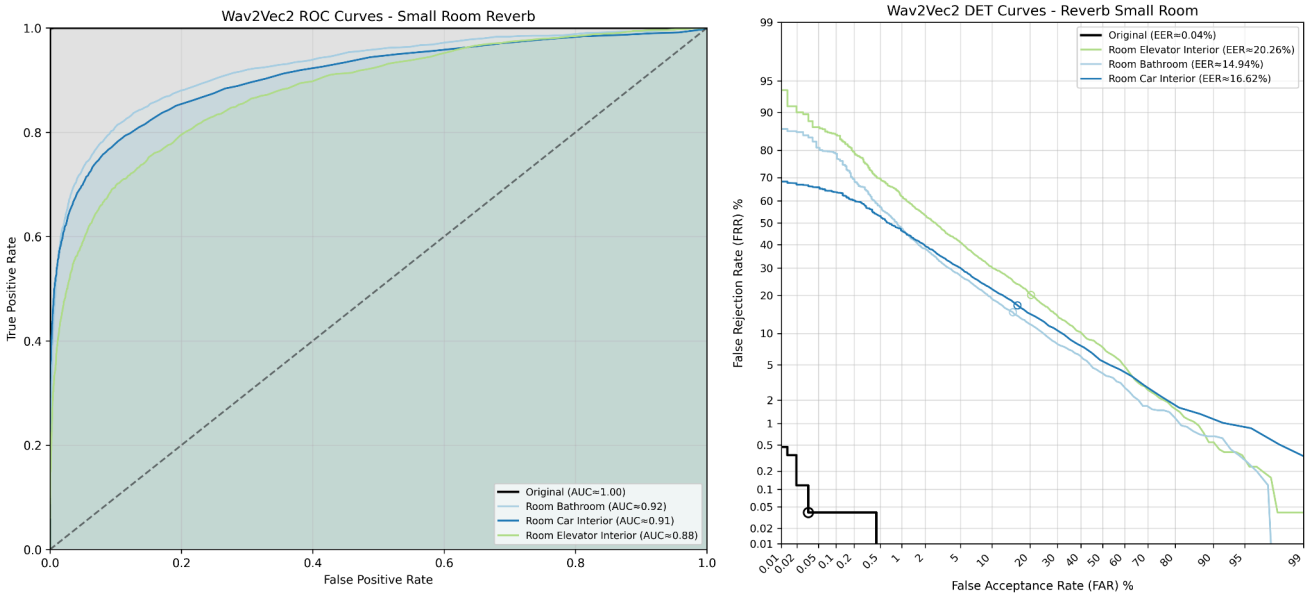

Figure 27: Wav2Vec2 ROC and DET Curves -Small Room Reverb

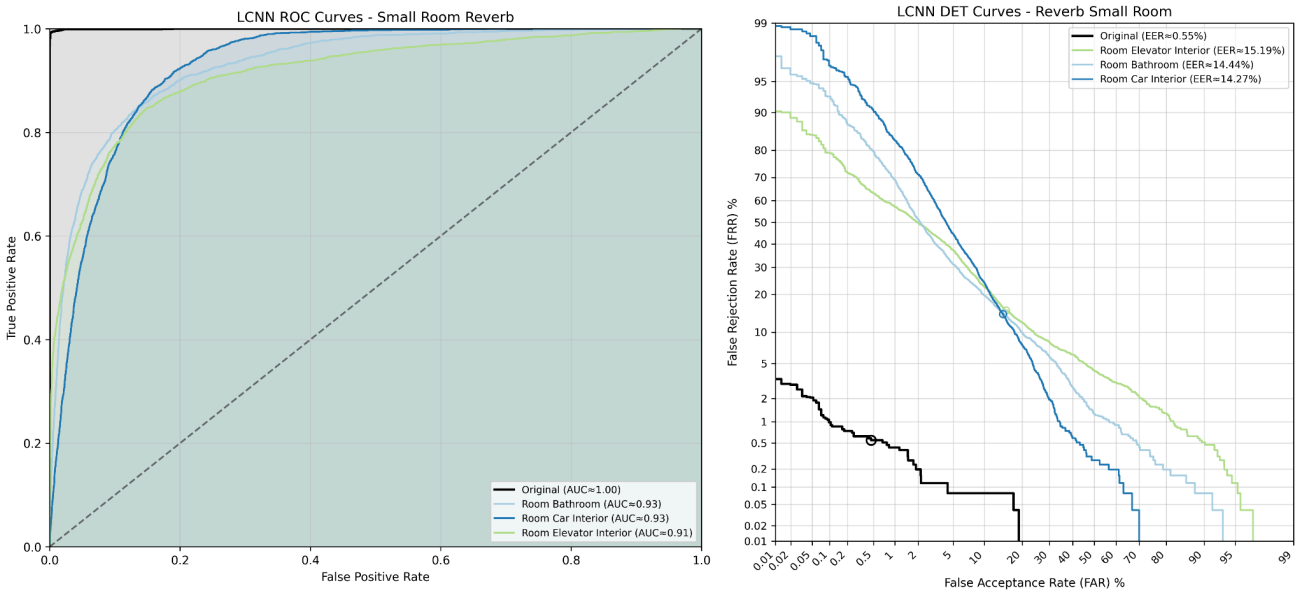

Figure 28: LCNN ROC and DET Curves -Small Room Reverb

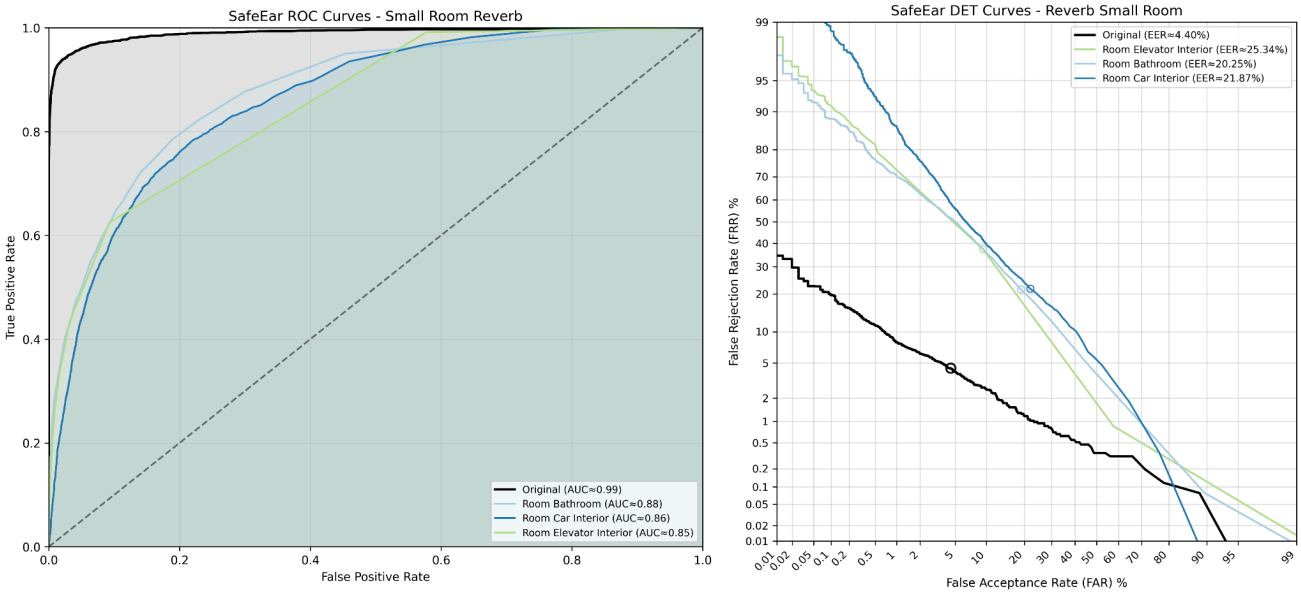

Figure 29: SafeEar ROC and DET Curves -Small Room Reverb

Looking at figure 27, Wav2Vec2 shows notable sensitivity to all 3 kinds of small room reverberations. Its ROC curves shift away from the top-left corner, with AUC dropping to 0.93 for the bathroom, 0.91 for the car interior, and 0.89 for the elevator interior. The DET curves also reveal high error rates, especially in the elevator condition (EER = 20.26%). Since all three environments have RTs ranging from 0.04 s to 0.34 s, it indicates that Wav2Vec2's performance degrades noticeably even with short RTs, having limited robustness under mild reverb.

LCNN, in Figure 28, demonstrates slightly better resilience in small room reverberation than Wav2Vec2. The ROC curves remain closer to the top-left, and its AUC values stay above 0.91 across all three IRs (0.93 for bathroom and car interior, and 0.91 for elevator). However, its DET curves still show elevated EERs: 14.44% (bathroom), 14.27% (car interior), and 15.19% (elevator). Therefore, LCNN is also affected by reverberations from closed small spaces.

SafeEar has the largest drop in small room reverbs. The ROC curves fall further than the other models, especially for the elevator and bathroom IRs. Its AUC drops to 0.85 for elevator, 0.88 for bathroom, and 0.86 for car interior. In addition, its DET curves show the highest error rates: 25.34% for elevator, 21.87% for car interior, and 20.25% for bathroom.

Overall, all models exhibit performance degradation, despite the relatively short reverberation times. Wav2Vec2 shows the steepest decline in accuracy, especially under the elevator IR. LCNN maintains slightly better robustness, with more stable AUC and EER values. SafeEar, however, experiences the most pronounced increase in error rates, indicating limited tolerance even to mild reverberation.

4.6.2 Large Room

Large room IRs are chosen from church, a room from a large palace (Elveden Hall), and an empty train station. These environments introduce sustained echoes that can significantly blur speech. The most reverberant in this group is Elveden Hall with an RT30 of 2.00 s, followed by the train station (1.72 s) and the baptist church (1.67 s).

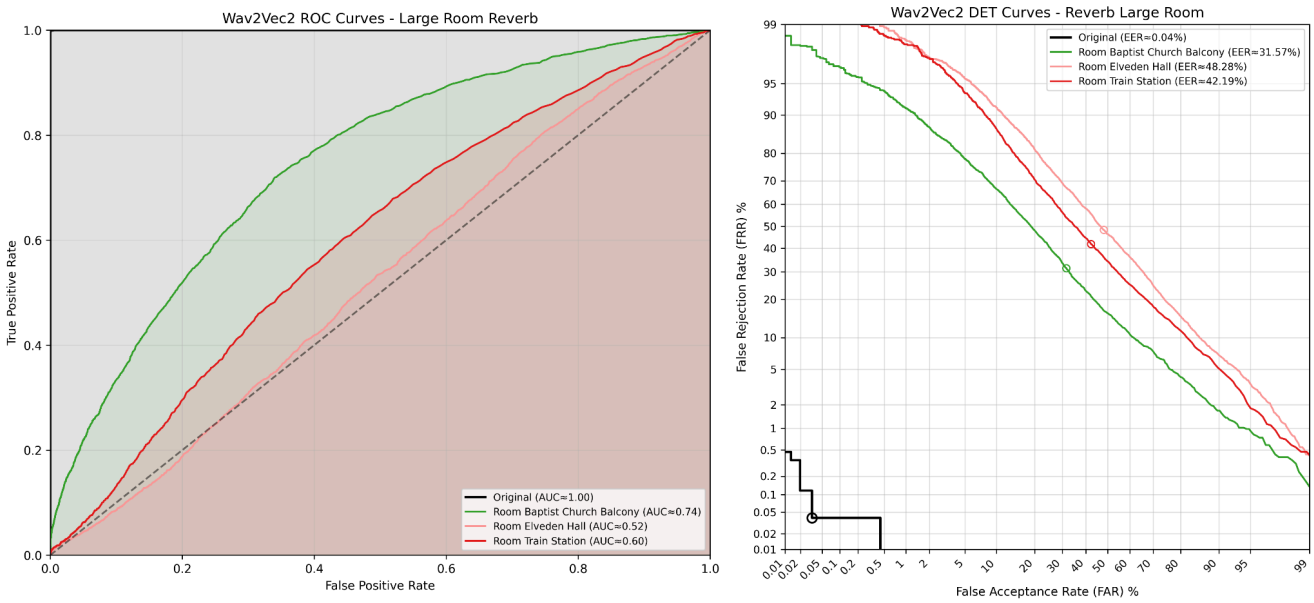

Figure 30: Wav2Vec2 ROC and DET Curves -Large Room Reverb

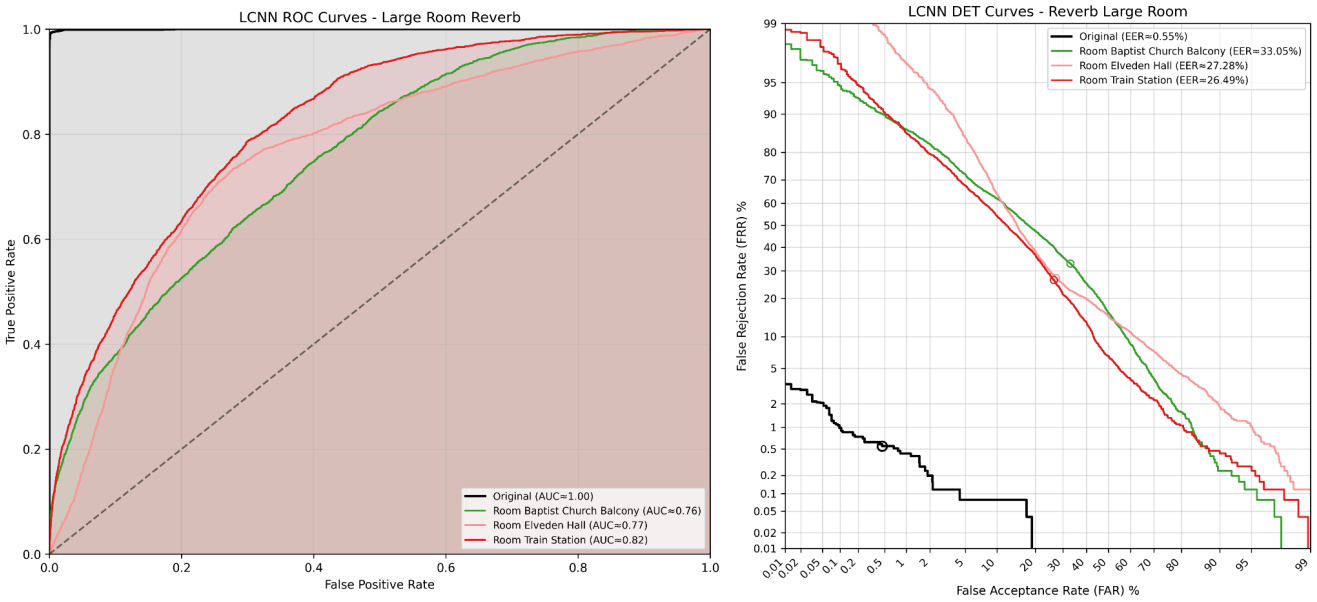

Figure 31: LCNN ROC and DET Curves -Large Room Reverb

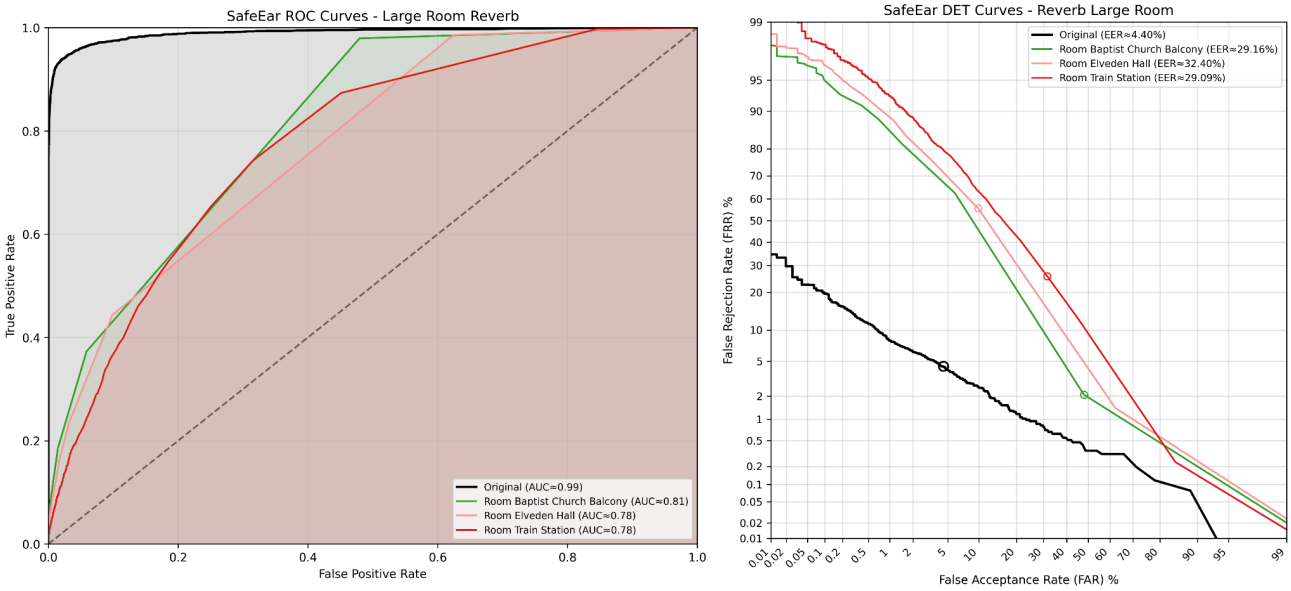

Figure 32: SafeEar ROC and DET Curves -Large Room Reverb

Under large room conditions where RT30 values range from 1.67 s to 2 s, Wav2Vec2 exhibits substantial performance degradation. The ROC curves shift significantly downward, with AUC dropping to 0.74 (church), 0.60 (train station), and as low as 0.52 in Elveden Hall. The degradation correlates with reverberation time—longer RT30 leads to poorer performance. In the most reverberant condition (Elveden Hall,

RT30 = 2 s), the model approaches random guessing. DET curves confirm this trend, with EER rising to 31.57%, 42.19%, and 48.28%, respectively, indicating major difficulty handling prolonged echoes.

LCNN also shows clear performance degradation under large room reverberation, though slightly less severe than Wav2Vec2. The ROC curves show noticeable downward shifts, with AUC values dropping to 0.76 (church), 0.82 (train station), and 0.77 (Elveden Hall). While still above random, these scores reflect significant impact from long RT30s. DET curves support this, with EERs increasing to 33.05%, 26.49%, and 27.28%, respectively.

SafeEar shows similar performance trends under large room reverb, with slightly higher AUC values than LCNN in some cases. Its ROC curves also decline, reaching 0.81 for church and 0.78 for train station and Elveden Hall. While the model maintains more stable classification accuracy compared to Wav2Vec2, performance still drops under prolonged reverberation. DET curves show EERs of 29.16%, 29.09%, and 32.40%, indicating that SafeEar also struggles in these environments

Overall, all three models experience substantial performance degradation due to prolonged reverberation. Wav2Vec2 is the most affected, with AUC dropping to near 0.5 in the most reverberant environment and EER rising above 48%, indicating near-random classification. LCNN and SafeEar also suffer a notable decline, although performing relatively better than Wav2Vec2.

4.6.3 Open Space

IRs in this category are from open environments including Koli national park, Abies grandis forest, and Haven beach from Puerto Rico. Among the three, the forest is the most reverberant in this group (0.58 s), followed by the beach (0.40 s) and Koli Park, which has the shortest RT30 (0.17 s). These IRs provide a diverse view of how models handle reverberation in less enclosed, real-world conditions.

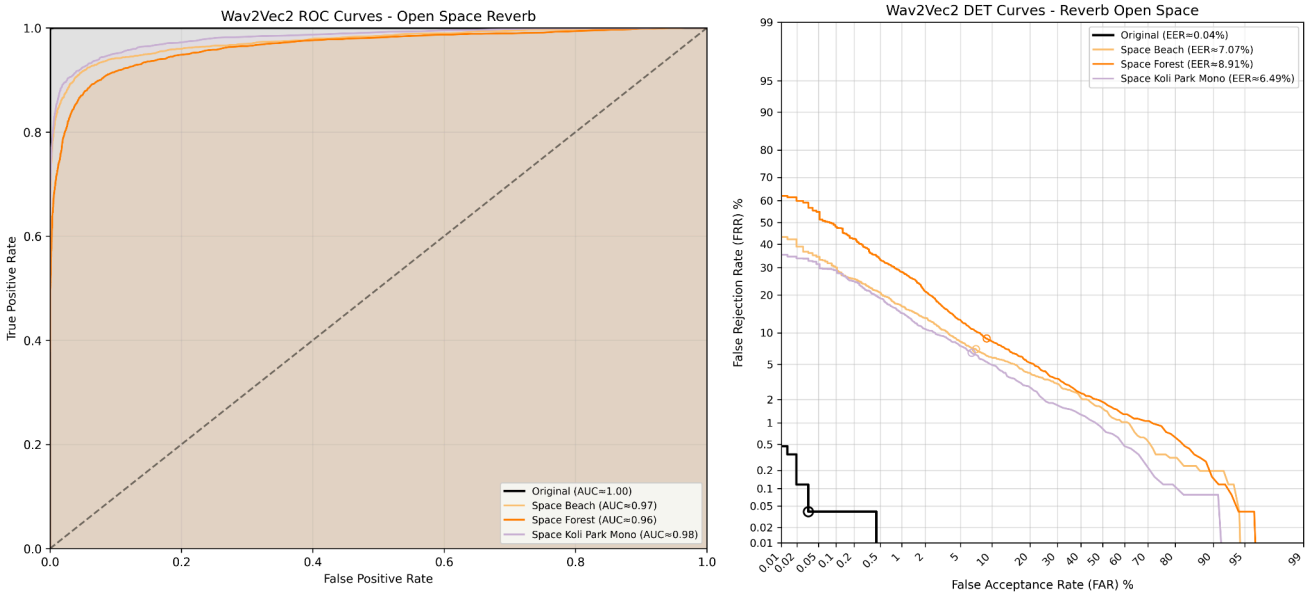

Figure 33: Wav2Vec2 ROC and DET Curves -Open Space Reverb

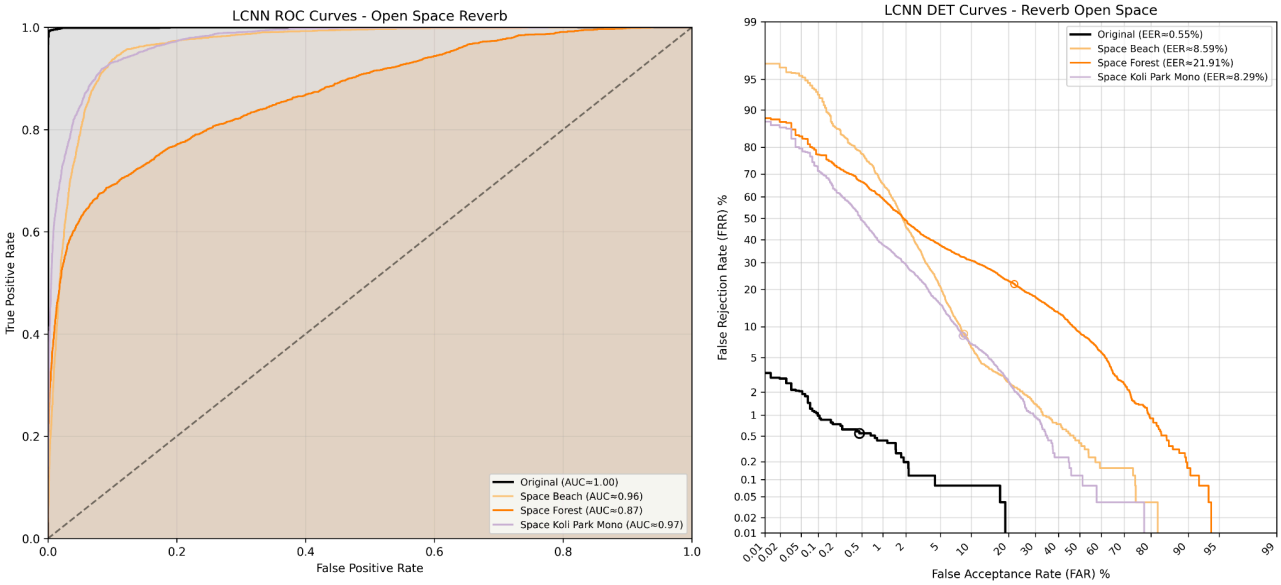

Figure 34: LCNN ROC and DET Curves -Open Space Reverb

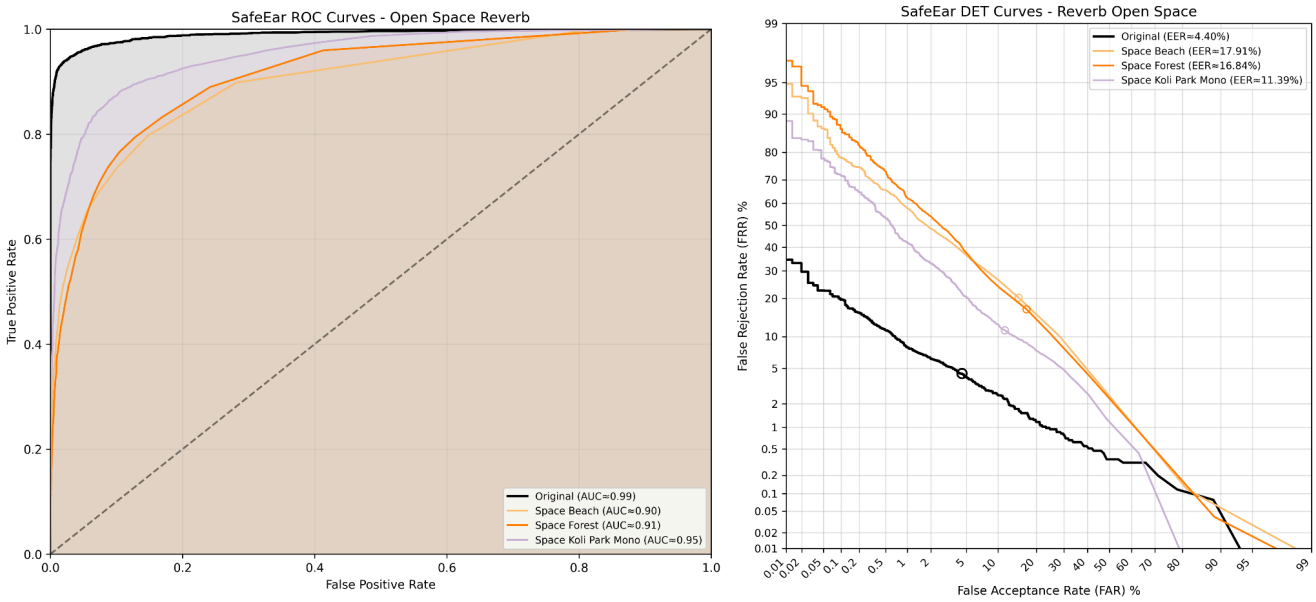

Figure 35: SafeEar ROC and DET Curves -Open Space Reverb

The open space IRs have longer RT30s (0.17 s – 0.58 s) on average than the small room IRs (0.04 s – 0.34 s) in general. Under this condition, Wav2Vec2 maintains strong performance under open space conditions. The ROC curves remain close to the top-left corner, and the AUC scores stay relatively high across all IRs: 0.97 (beach), 0.96 (forest), and 0.98 (Koli Park). Despite some elevation in the DET curves, particularly for the forest (EER = 8.91%) and beach (EER = 7.07%), the overall degradation is minor.

LCNN shows moderate sensitivity to open-space reverberation. The ROC curves slightly shift downwards, with AUC values ranging from 0.87 (forest) to 0.97 (Koli Park). The DET curves indicate elevated EERs, especially under the forest IR (EER = 21.91%), which appears to challenge the model more than other open environments. While LCNN retains relatively strong performance in beach and park conditions, its performance drops more noticeably in the forest setting.

SafeEar also experiences performance loss among the three models under open space conditions, slightly worse than LCNN. The ROC curves display a more pronounced downward shift, and AUC values drop to 0.90 (beach), 0.91 (forest), and 0.95 (Koli Park). The DET curves reveal high EERs, particularly 17.91% for the beach and 16.84% for the forest.

Overall, although open space IRs have moderately longer RT30s than small rooms, model performance remains relatively stable overall. Wav2Vec2 demonstrates

strong robustness, with minimal degradation in both ROC and DET curves. LCNN shows mild to moderate sensitivity, with performance particularly affected by the forest IR, which yields the highest EER. SafeEar is the most impacted in open space conditions among the three, showing the lowest AUCs and the highest EERs among the three models.

4.6.4 Summary for Reverberation Augmentation

Across all reverb conditions, large room reverberation causes the most significant performance degradation for all three models, with sharp drops in AUC and substantial increases in EER. Small room and open space reverbs have a milder impact by comparison. Wav2Vec2 is highly sensitive to large rooms but remains strong in open spaces. LCNN is relatively stable under small and open environments, though performance drops under prolonged echoes. SafeEar shows consistent mid-level performance but is more affected by both large room and open space IRs. Overall, the severity of reverberation, particularly longer RT30s in large spaces, has the strongest influence on model robustness.

5. DISCUSSION & INTERPRETATION

The previous section presented how the three models—LCNN, Wav2Vec2, and SafeEar—performed under different types of audio distortion, including noise, codec compression, time-stretching, pitch-shifting, and reverberation. The results reveal clear differences in robustness, degradation patterns, and generalization capacity, depending on the distortion type and model architecture. Building on these findings, this following discussion section shifts focus to interpreting the significance of these results. This includes exploring models' behaviors based on their architecture and highlighting unexpected trends that emerged during testing. By interpreting these findings, we can better understand how deepfake detection models work in real-world conditions and how they might be improved in the future.

5.1 Key Results Interpretation

5.1.1 Noise-Based Augmentations

Among LCNN, Wav2Vec2, and SafeEar, there was a consistent trend across all three types of noise (white, pink, and brown) observed from the result section above: as the Signal-to-Noise Ratio (SNR) decreased, model performance declined. This finding aligns with general observations in speech processing, where lower SNRs inherently make the target signal harder to distinguish from noise, leading to performance degradation across various models and tasks (Reddy et al., 2019). However, each model demonstrates a specific SNR threshold where performance begins to deteriorate sharply, which provides insight into their spectral vulnerabilities.

Wav2Vec2 is the most robust model across all three noise types. Its performance remains stable under white and pink noise down to 15 dB SNR, with the critical degradation threshold occurring at 10 dB. Below this point, performance declines rapidly. Under brown noise, however, Wav2Vec2 shows no critical threshold at all; it remains consistently accurate even at 0 dB SNR. This suggests that the model's learned waveform representations are particularly resilient when mid- and high-frequency components of speech are preserved, as is the case with brown noise. This is aligned with the findings of Zhu et al. (2022) that Wav2Vec2.0 performs well in noisy environments and across different domains due to its self-supervised pre-training approach that enables it to learn more noise-invariant speech representations.

LCNN is the most sensitive model to noise overall, especially white and pink noise. The first clear signs of degradation appear as early as 30 dB SNR, with 25 dB marking the critical threshold where performance becomes meaningfully impaired. This early vulnerability reflects the model's dependence on LFCC features, which extract information from the spectral envelope (Zhou et al., 2011). Because white and pink noise effectively mask the mid-to-high frequency bands where speech information mainly present (Licklider, 1948), LCNN's performance is quickly compromised. In contrast, LCNN is more stable under brown noise, where the masking is concentrated in the low-frequency range. It maintains usable performance down to 15 dB SNR, with noticeable degradation beginning only below 10 dB.

SafeEar shows a hybrid behavior. Under both white and pink noise, its robustness is better than LCNN's in low-SNR extremes, but with a similar degradation pattern. While some instability appears at higher SNRs, the most critical shift occurs at 15 dB SNR, where error rates increase sharply. In brown noise conditions, SafeEar's performance is the weakest of the three models, with degradation beginning gradually at 15 dB and becoming more severe below 10 dB. This behavior can be attributed to its architecture: SafeEar discards semantic tokens and relies solely on acoustic tokens that encode prosodic and timbral features (Li et al., 2024b). Brown noise, with its overwhelming energy in the low-frequency domain, directly interferes with these cues, making the acoustic tokens less reliable (Garnier & Henrich, 2014).

From the results, it is clear to see that models that process raw waveforms directly, such as Wav2Vec2, can achieve better noise robustness than systems built on fixed frequency-domain features like LFCC. This is not because frequency-based features are inherently flawed, but because they are based on fixed, non-adaptive transformations (such as FFT, filterbanks, and log energy), which discard important information such as phase and fine temporal details (Dua et al., 2023). When additive noise corrupts the energy in frequency bins, the fixed feature extraction pipeline cannot compensate, and since much of the raw signal has already been abstracted or compressed, the model has fewer redundant cues to rely on. In contrast, raw audio models like Wav2Vec2 and SafeEar maintain access to the full waveform, including phase and timing information (Baevski et al., 2020)(Li et al., 2024b). Therefore, they are able to extract more resilient representations, especially in the presence of complex or broadband noise.
While this advantage depends on architecture and training conditions, the results in this study highlight that end-to-end raw audio models have greater potential for generalizing under noise distortions.

In summary, Wav2Vec2's robustness is driven by its ability to learn meaningful, noise-tolerant representations from the raw waveform. LCNN's vulnerability to white and pink noise reflects how fragile fixed frequency-domain features can be under broadband noise. SafeEar's specific decline under brown noise stems from its architectural dependence on low-frequency acoustic encoding. These results show that a model's noise robustness depends not just on whether it uses raw audio or frequency features, but on

which parts of the signal it focuses on and what information it keeps or discards during feature extraction.

5.1.2 Codec-Based Augmentation

Across all tested lossy codecs, performances of three models remain largely stable. Both Wav2vec2 and LCNN have both AUC and EER values that stay close to those observed under clean conditions. This suggests that the key acoustic features used by these models are mostly preserved after typical audio compression, making them robust to codec-induced distortions.

SafeEar is slightly influenced by codec compression more, especially under GSM codec. GSM codec is typically used in low-bitrate mobile telephony and introduces stronger compression artifacts. These artifacts may interfere with SafeEar's acoustic tokenization process, but the overall impact of codec compression on SafeEar was still modest.

In summary, codec compression was the least disruptive form of distortion tested in this study. All three models, especially Wav2Vec2 and LCNN, maintained strong performance across various codec formats, indicating that detection systems are generally resilient to common audio compression techniques used in communication and storage. This trend does not align with prior findings from Cohen et al. (2022), which noted that codec compression significantly degraded performance of audio deepfake detection systems such as ResNet, SENet, and OCS-ResNet.

5.1.3 Time Stretch Augmentation

From the results in the analysis section, it is clear that time-stretching impacts all models, but the degree and nature of degradation vary significantly depending on the input representation. In general, slowing down the audio (tempo < 1.0) caused more severe degradation than speeding it up (tempo > 1.0), particularly for models relying on raw waveform features.

Wav2Vec2, which operates directly on raw waveforms (Baevski et al., 2020), is highly sensitive to changes in tempo, especially when slowing down. Its performance remains stable down to a stretch factor of 0.6×, but 0.5× marks a critical threshold where degradation becomes abrupt. Below this point, AUC drops quickly and EER rises steeply,

exceeding 40% at 0.4× and deteriorating further at more extreme values. For fast tempo conditions, Wav2Vec2 stays reliable up to 1.7×, but performance begins to degrade noticeably at 1.8×, with sharp declines continuing through 1.9× and 2.0×. These patterns suggest that the Wav2Vec2's learned features heavily depend on natural temporal dynamics, and that both slowing down and speeding up beyond these thresholds disrupt the signal beyond what Wav2Vec2 can generalize.

In contrast, LCNN remains remarkably stable across all stretch factors. It maintains near-perfect AUC and low EER across the full tempo range from 0.1× to 2.0×, showing no clear threshold of degradation. This robustness is due in part to its use of LFCC features, which are based on short-term spectral analysis and offer strong invariance to tempo changes (Zhou et al., 2011). Since frame-level spectral content remains largely consistent despite changes in global tempo, LCNN is able to maintain high detection accuracy even under extreme time-stretching.

SafeEar falls between the other two models but shows strong overall robustness. When slowing down the audio, AUC values remain above 0.90 even at 0.1×, and the degradation is smooth and gradual with no sharp collapse. While there is no distinct threshold, mild performance softening begins around 0.4×–0.3×, suggesting a boundary where the model's acoustic tokens begin to drift from typical prosodic patterns. For fast tempo conditions, SafeEar performs even better. Its AUC stays above 0.95 across all speed-up factors from 1.1× to 2.0×, with EER remaining low. There is no critical threshold identified for fast tempo; the model's performance remains consistently strong throughout.

Since the time stretch method is performed by SoX, which uses phase vocoder-based methods like WSOLA (Waveform Similarity Overlap-Add) , the asymmetric effect between slowing down and speeding up can be explained by the way these algorithms manipulate the signal (SoX Development Team, 2016). When slowing down audio, the algorithm must insert or synthesize additional frames to extend the duration without altering pitch. This process involves estimating phase information to maintain continuity between frames, which is particularly challenging. Imperfect phase alignment can introduce artifacts such as smearing of transients and phase distortion

(Juillerat, 2017), which severely affect models like Wav2Vec2 that rely on precise time-domain waveform structures.

Furthermore, slowing down the audio increases the duration of individual phonemes, which alters the natural timing, rhythm, and prosodic structure of speech (Driedger & Müller, 2016). For models like SafeEar, which use acoustic tokens that capture speech timing and intonation (Li et al., 2024b), these unnatural prosodic changes may result in token sequences that deviate from typical patterns, leading to reduced classification accuracy. In contrast, speeding up the audio usually involves discarding frames rather than generating new ones. Although this results in some loss of information, it maintains the original phase relationships within the remaining frames, introducing fewer waveform distortions (Driedger & Müller, 2016). As a result, the impact of speeding up is generally less severe across all models, especially to those sensitive to temporal structure.

5.1.4 Pitch Shift Augmentation

Similar to time-stretch augmentation, the results for pitch shifting augmentation also suggest that pitch shifting impacts all three models, though the severity and pattern of degradation vary depending on model architecture and input representation. In general, larger pitch shifts, both upward and downward, lead to greater performance degradation, but each model demonstrates distinct tolerance thresholds.

Wav2Vec2 is the most sensitive model under pitch shifts. Its performance remains stable up to ±5 semitones, but a clear drop begins at ±6 semitones, where AUC declines and EER increases sharply. Beyond this point, especially from ±7 to ±10 semitones, the model deteriorates significantly, with EER exceeding 30%. This sensitivity is due to its reliance on raw waveform input, which includes precise timing, phase, and spectral structures (Baevski et al., 2020). When SoX applies pitch shifts using resampling and phase vocoder techniques, these structures may become distorted. Artifacts such as transient smearing, phase misalignment, and distortion of harmonic structure, including changes in the relative strength and spacing of overtones, disrupt the signal's temporal and spectral cues (Juillerat, 2017). These are important components that Wav2Vec2's

learned representations rely on. This explains the steep performance drop after the ±6 semitone threshold.

LCNN, by contrast, is the most robust model under pitch shift augmentation. It shows only gradual and symmetric performance decline across both upward and downward shifts. AUC remains above 0.90 and EER increases slowly even at ±10 semitones, with mild degradation becoming more visible around ±6 semitones. This robustness comes from LCNN's use of LFCC features, which focus on the overall shape of the spectrum and are less sensitive to changes in pitch. Because these features discard fine frequency detail and pitch information, LCNN maintains reliable performance even when the signal's fundamental frequency is altered (Zhou et al., 2011).

SafeEar demonstrates a moderately stable response to pitch shifts, but with a noticeable asymmetry: it is particularly vulnerable to downward shifts. It handles the upward pitch shifts well with AUC remaining above 0.95 and no sharp threshold of failure. However, for downward pitch shifts, its performance begins to decline around –4 semitones, and worsens steadily as the pitch drops further. At –6 semitones and beyond, the model shows significantly higher EERs, reaching 34% at –10 semitones. This asymmetric degradation likely reflects how pitch shifting affects SafeEar's acoustic token representation. SafeEar discards semantic tokens and relies entirely on acoustic tokens that capture prosody and timbre (Li et al., 2024b). Lower-pitched signals, which are often produced with resampling artifacts and low-frequency distortions (Juillerat, 2017), can disrupt the tokenization process. These artifacts may interfere with the acoustic structure that SafeEar's detector depends on, resulting in lower classification accuracy.

Overall, pitch shifting introduces distortions that affect all models, but to different extents and with different patterns. Similar to time-stretch augmentation, fixed frequency-domain features like LFCC are less affected by manipulations targeting the fundamental frequency (F0) and the precise time-domain waveform structure. Wav2Vec2 exhibits the clearest threshold of failure at ±6 semitones, while LCNN shows gradual degradation without a sharp drop, and SafeEar performs well with upward shifts but begins to decline early under downward shifts. These results emphasize that how each model extracts and processes pitch-affected features, whether through fixed spectral

shapes, learned waveform patterns, or tokenized prosody, plays a critical role in determining its sensitivity to pitch-based audio modifications.

5.1.5 Reverb-based Augmentation

Reverberation was found to significantly degrade model performance, particularly when reverberation time (RT30) was long or when the acoustic environment introduced complex reflection patterns. This degradation is probably due to temporal smearing and spectral distortion caused by overlapping reflections, which obscure the fine-grained features needed for accurate spoof detection.

Across all environments, Wav2vec2 is the most vulnerable to reverberations, especially in large rooms with long and dense reverberations. Despite excelling in clean and noisy conditions, its performance dropped sharply in highly reverberant environments, with EERs exceeding 40% in some cases. This suggests that Wav2Vec2's reliance on raw waveform input makes it highly sensitive to time-domain and phase-related distortions that disrupt its learned feature representations.

In contrast, LCNN demonstrated the best resilience in these conditions, particularly under extreme reverberation. Although performance declined with longer RT30, the model retained lower EERs than Wav2Vec2 in large rooms. This robustness likely stems from its use of LFCC features (a frequency-based feature), which are known to be less sensitive to convolutional distortions and may suppress the impact of long-term echoes due to their cepstral processing (Zhou et al., 2011). Interestingly, LCNN's performance in small and open spaces was more affected. This may suggest that other reverberation characteristics other than RT30 played a role. Some possible factors may be specific spectral coloration introduced by the room modes, density and timing of the early reflections, or the severity of comb filtering effects (Ko et al., 2017).

SafeEar, which also takes raw audio as input like Wav2Vec2, demonstrates moderate degradation across all reverb types. While it performed worse than LCNN in most cases, it was more stable than Wav2Vec2 under heavy reverberation. This may be because SafeEar's encoder relies less on temporal precision compared to the encoder of Wav2Vec2. However, its overall error rates remained higher, suggesting that while SafeEar's encoder may be less sensitive to temporal smearing, the acoustic token

representations it generates are still noticeably impacted by reverberation. SafeEar's ability to function correctly depends heavily on its acoustic tokens accurately capturing essential sound characteristics, such as timbre (related to spectral qualities) and prosody (timing and pitch patterns) (Li et al., 2024b). Meanwhile, reverberation directly interferes with these acoustic foundations (Ko et al., 2017). Therefore, the distortion of these essential timbre and prosody features by reverberation leads to the generation of less accurate or consistent acoustic tokens, ultimately degrading the backend detector's performance as it relies on these compromised representations to identify spoofing cues.

The analysis and interpretation above reveals that reverberations may degrade model performance through effects like temporal smearing and general spectral distortion. Among these effects, one specific form of spectral distortion, known as sharp nulls or dips in the frequency response of reverberant impulse responses (IRs), may play a particularly important role. These deep frequency notches, often resulting from comb filtering, represent areas of significant signal loss and may eliminate important information that detection models rely on (Sadjadi & Hansen, 2014). To investigate the impact of these spectral nulls more closely, an additional experiment was carried out and is detailed in the section below.

5.1.5.1 Hypothesis Verification

To evaluate whether the dips in the frequency spectrum of impulse responses (IRs) contributed to the decline in model performance, a modified version of the small room IRs was created. This was done by applying a moving average smoothing technique to the magnitude of the IRs. The aim was to reduce the sharpness of these spectral dips while preserving the overall shape of the frequency response.

5.1.5.2 Method of Smoothing the Impulse Response

Smoothing was carried out using Python libraries, specifically NumPy and SciPy. To begin, each impulse response (IR) from the small room condition was converted into the frequency domain using the Fast Fourier Transform (FFT). The magnitude spectrum was then smoothed by convolving it with a uniform kernel of adjustable size, while the phase spectrum was kept unchanged to preserve temporal alignment. An inverse FFT was applied to return the smoothed result to the time domain. Three window sizes (5, 15, and

30) were chosen to explore varying levels of smoothing. Each smoothed IR was used to generate reverberated audio by convolving it with the clean speech samples. The energy of the resulting audio was normalized to match the original.

To assess model performance, Wav2Vec2, LCNN, and SafeEar were evaluated on these smoothed datasets. Performance metrics including ROC curves, DET curves, AUC, and EER were computed again and visualized using Matplotlib. For each model and IR, the smoothed IR version that yielded the lowest EER was selected as the best smoothed configuration to do the comparison with the non-smoothed small room IR augmented dataset.

5.1.2.2 Results & Interpretation of Comparing Smoothed & Non-Smoothed IRs

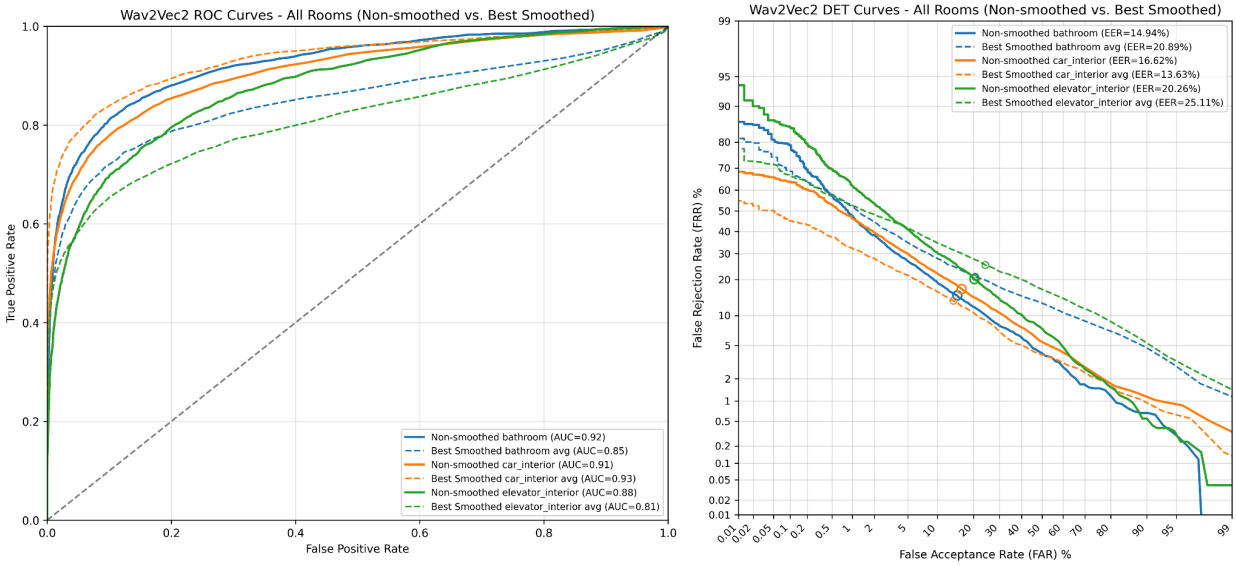

Figure 36: Wav2Vec2 ROC and DET Curves - IR Comparisons

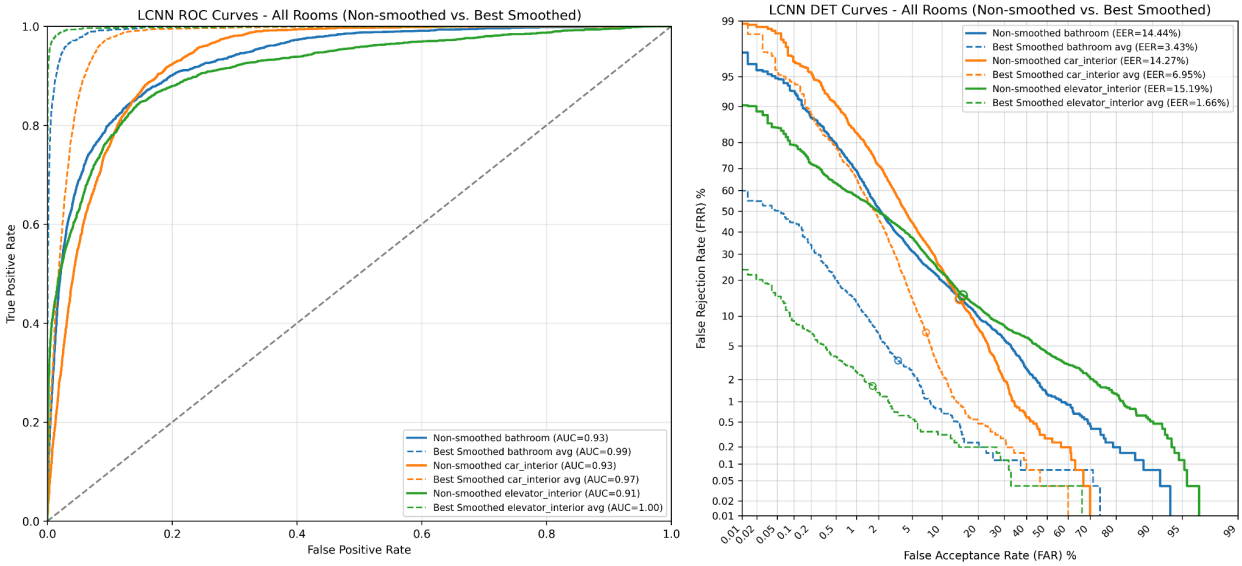

Figure 37: LCNN ROC and DET Curves - IR Comparisons

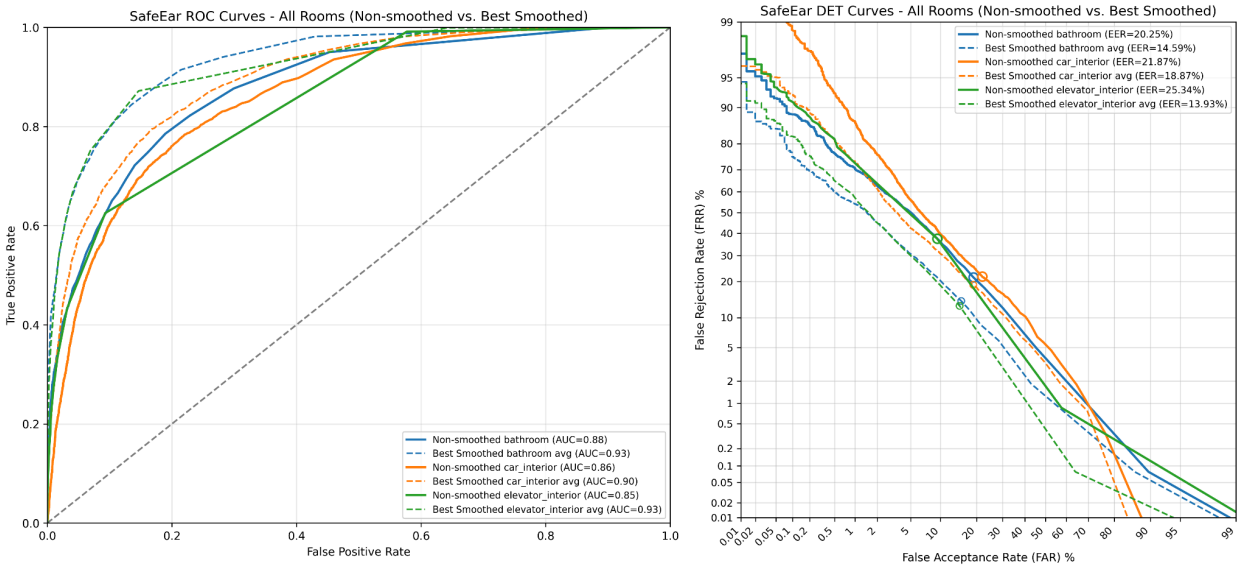

Figure 38: SafeEar ROC and DET Curves - IR Comparisons

Figure 36 to Figure 38 show the ROC and DET curves for Wav2Vec2, LCNN, and SafeEar, respectively. Each figure compares model performance using non-smoothed IRs (solid lines) and their corresponding best-performing smoothed IRs (dashed lines) for three small room environments: bathroom, car interior, and elevator interior.

Looking at Figure 36 for Wav2Vec2, the ROC curves for smoothed IRs are interweaving between the ROC curves for smoothed IRs, indicating inconsistent effects

from smoothing. In the car interior condition, the smoothed curve rises above the original, suggesting improved performance, as confirmed by a reduced EER (16.62% to 13.63%). However, in the bathroom and elevator conditions, the smoothed ROC curves shift downward and the DET curves rise, with EER increasing to 20.89% and 25.11%, respectively. This suggests that Wav2Vec2's performance is negatively affected by smoothing in most cases, possibly because the model relies on subtle time-domain or spectral cues that are distorted or removed by smoothing the IR magnitude response.

In contrast, Figure 37 demonstrates a clear and consistent improvement for LCNN across all environments. The smoothed ROC curves are closer to the top-left corner, and the DET curves are significantly lower, indicating better discrimination at all thresholds. The EER values drop sharply, with the elevator interior condition improving from 15.19% to just 1.66%. These results support the hypothesis that sharp spectral dips degrade the effectiveness of LFCC features, and that smoothing helps restore useful spectral information.

Figure 38 shows moderate but consistent gains for SafeEar. All smoothed ROC curves are higher than their original counterparts, and the DET curves show lower false rejection and acceptance rates. The largest gain appears in the elevator environment, where the EER improves from 25.34% to 13.93%. This suggests that while SafeEar's acoustic token representation is less sensitive to fine spectral detail than LFCCs, it still benefits from a cleaner and more balanced spectral input.

In summary, although IR smoothing leads to clear improvements for LCNN and SafeEar and mixed results for Wav2Vec2, these findings alone do not conclusively demonstrate that sharp spectral dips are the primary factor behind performance degradation. Other factors, such as phase distortion, temporal smearing, or model-specific sensitivities, may also contribute. Further analysis is needed to isolate the specific role of these spectral nulls in model performance.

5.2 Future Work

One possible direction for future research is to explore alternative methods for applying temporal augmentations, particularly time-stretching and pitch-shifting. This thesis implemented these augmentations using SoX. It is a more general-purpose tool which relies on phase vocoder-based processing and may introduce artifacts such as

transient smearing and phase inconsistencies. These artifacts can significantly influence model behavior, especially for raw waveform models like Wav2Vec2. Future work could implement similar augmentations using Rubberband, a high-quality library designed for more precise time–frequency transformations. Rubberband has been widely used in music production due to its better handling of complex audio content and transient preservation. Comparing augmentations produced by SoX and Rubberband would help identify whether the differences in model robustness stem from the distortion method or from the algorithmic characteristics of the tools used.

Another important direction is to deepen the investigation into why reverberation affects model performance. This study extended to test on using a smoothing method to reduce spectral dips in small room impulse responses to see if sharp notches were a key factor. However, more research is needed to explore other acoustic characteristics such as reflection density and spectral coloration might also affect model performance. Beyond small rooms, it is also important to extend this analysis to more complex reverberant environments, such as large halls and open spaces, where sound reflections are more diffuse and temporally dispersed. Investigating these conditions could reveal additional limitations in current detection models and inform the development of architectures or preprocessing techniques that are more resilient to realistic acoustic distortions.

In the end, since the codec compression reveals different results than some prior works, it may be worth looking into and examining more variety of bite rate for compression to further understand the impact of compression on audio deepfake detection systems.

6. CONCLUSIONS

This thesis investigated how real-world audio distortions affect the performance of deepfake detection models. While deepfake detection systems have been improved and achieved strong performance under clean conditions, their real-world reliability remains uncertain due to the presence of common audio degradations such as additive noise, codec compression, temporal distortions, and reverberation. Addressing this gap, the study introduced a comprehensive augmentation pipeline and evaluated three representative models: LCNN, Wav2Vec2, and SafeEar, across a wide range of controlled distortion scenarios.

The results show that each model responded differently to distortion types, depending on its architectural design and input representation. Wav2Vec2, which processes raw audio waveforms, showed high robustness under noise and codec compression, especially brown noise, but degraded quickly under heavy reverberation and temporal changes. LCNN, based on fixed frequency-domain features, demonstrated strong resistance to time-stretch and pitch-shift augmentations, but was highly sensitive to noise, particularly in the mid- and high-frequency range. SafeEar, a privacy-preserving model using acoustic tokenization, performed moderately across all distortions but showed limited generalization under downward pitch shifts and reverberation. A follow-up experiment on impulse response smoothing confirmed that sharp spectral notches may play a role in degrading model performance, especially for LCNN and SafeEar.

These findings suggest that no single architecture currently offers universal robustness across all real-world distortions. Instead, each model exhibits strengths and vulnerabilities shaped by its approach to feature extraction and input representation. The results emphasize the need for distortion-aware training and evaluation in the development of future detection systems. Moreover, the inclusion of SafeEar in this study highlights an emerging research direction in privacy-aware audio detection, though its performance trade-offs under distortion require further exploration. Overall, this work contributes a comprehensive assessment of distortion impacts and provides a foundation for building more reliable and generalizable detection systems.

REFERENCES

ASVspoof Challenge. (2021). *LA Baseline-LFCC-LCNN*. GitHub. https://github.com/asvspoof-challenge/2021/tree/main/LA/Baseline-LFCC-LCNN

Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). Wav2vec 2.0: A framework for self-supervised learning of speech representations. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 12449–12460). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf

Chesney, R., & Citron, D. K. (2019). Deep fakes: A looming challenge for privacy, democracy, and national security. *California Law Review*, *107*(6), pp.1753-1820. https://doi.org/10.2139/ssrn.3213954

Cohen, A., Rimon, I., Aflalo, E., & Permuter, H. (2022). A study on data augmentation in voice antispoofing. *Speech Communication*, *141*, pp.56-67. https://doi.org/10.1016/j.specom.2022.04.005

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. *Association for Computing Machinery*, pp. 233–240. https://doi.org/10.1145/1143844.1143874

Driedger, J., & Müller, M. (2016). A review of time-scale modification of music signals. *Applied Sciences*, *6*(2), pp.57. https://doi.org/10.3390/app6020057

Dua, M., Akanksha, N., & Dua, S. (2023). Noise robust automatic speech recognition: Review and analysis. *International Journal of Speech Technology*, *26*(2), 475–519. https://doi.org/10.1007/s10772-023-10033-0

Facebook Research. (2017). *fairseq/examples/wav2vec* . GitHub. https://github.com/facebookresearch/fairseq/tree/main/examples/wav2vec#wav2vec-20

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Freesound. (2012). Freesound. https://freesound.org/

Garnier, M., & Henrich, N. (2014). Speaking in noise: How does the lombard effect improve acoustic contrasts between speech and ambient noise? *Computer Speech & Language*, *28*(2), 580–597. https://doi.org/10.1016/j.csl.2013.07.005

Juillerat, N. (2017). Audio time stretching with controllable phase coherence. *Journal of the Audio Engineering Society*, *142*, pp.9780. https://aes2.org/publications/elibrary-page/?id=18656

Kamble, M. R., Sailor, H. B., Patil, H. A., & Li, H. (2020). Advances in antispoofing: From the perspective of asvspoof challenges. *APSIPA Transactions on Signal and Information Processing*, *9*, e2. Cambridge Core. https://doi.org/10.1017/ATSIP.2019.21

Kaneko, T., & Kameoka, H. (2018). CycleGANVC: Nonparallel voice conversion using cycleconsistent adversarial networks. *2018 26th European Signal Processing Conference (EUSIPCO)*, pp.2100-2104. https://doi.org/10.23919/EUSIPCO.2018.8553236

Khan, A., Malik, K. M., Ryan, J., & Saravanan, M. (2023). Battling voice spoofing: A review, comparative analysis, and generalizability evaluation of stateoftheart voice spoofing counter measures. *Artificial Intelligence Review*, *56*(1), pp.513-566. https://doi.org/10.1007/s10462023105398

Ko, T., Peddinti, V., Povey, D., & Khudanpur, S. (2015). Audio augmentation for speech recognition. *Interspeech 2015*. https://doi.org/10.21437/interspeech.2015-711

Ko, T., Peddinti, V., Povey, D., Seltzer, M. L., & Khudanpur, S. (2017). A study on data augmentation of reverberant speech for robust speech recognition. *International Conference on Acoustics, Speech, and Signal Processing*, pp.5220-5224. https://doi.org/10.1109/icassp.2017.7953152

Lavrentyeva, G., Novoselov, S., Tseren, A., Volkova, M., Gorlanov, A., & Kozlov, A. (2019). STC antispoofing systems for the asvspoof2019 challenge. *Interspeech 2019*. https://doi.org/10.21437/interspeech.2019-1768

Li, X., Li, K., Zheng, Y., Yan, C., Ji, X., & Xu, W. (2024a). *GitHub - letterligo/safeear: [ACM CCS'24] safeear: Content privacy-preserving audio deepfake detection*. GitHub. https://github.com/LetterLiGo/SafeEar?tab=readme-ov-file

Li, X., Li, K., Zheng, Y., Yan, C., Ji, X., & Xu, W. (2024b). SafeEar: Content privacy-preserving audio deepfake detection. *Association for Computing Machiner*, pp.3585-3599. https://doi.org/10.1145/3658644.3670285

Licklider, J. C. R. (1948). The influence of interaural phase relations upon the masking of speech by white noise. *The Journal of the Acoustical Society of America*, *20*(2), pp.150-159. https://doi.org/10.1121/1.1906358

Martin, A. F., Doddington, G. R., Kamm, T., Ordowski, M., & Przybocki, M. A. (1997). The DET curve in assessment of detection task performance. *Fifth European Conference on Speech Communication and Technology*, 1895–1898. https://doi.org/10.21437/EUROSPEECH.1997-504

Mirsky, Y., & Lee, W. (2021). The creation and detection of deepfakes: A survey. *ACM Computing Surveys (CSUR)*, *54*, 1. https://doi.org/10.1145/3425780

Müller, N. M., Pizzi, K., & Williams, J. (2022). Human perception of audio deepfakes. *Association for Computing Machinery*, pp.85-91. https://doi.org/10.1145/3552466.3556531

Müller, N., Czempin, P., Diekmann, F., Froghyar, A., & Böttinger, K. (2022). Does audio deepfake detection generalize? *Interspeech 2022*, 2783--2787. https://doi.org/10.21437/interspeech.2022-108

Murphy, D. T., & Shelley, S. (2010). OpenAIR: An interactive auralization web resource and database. *Journal of the Audio Engineering Society*, *8226*. https://aes2.org/publications/elibrary-page/?id=15648

Oord, A. van den , Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *Arxiv*. https://doi.org/10.48550/arXiv.1609.03499

Park, D., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E., & Le, Q. (2019). SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*. https://doi.org/10.21437/Interspeech.2019-2680

Reddy, C. K., Beyrami, E., Pool, J., Cutler, R., Srinivasan, S., & Gehrke, J. (2019). A scalable noisy speech dataset and online subjective test framework. *Arxiv*. https://doi.org/10.21437/interspeech.2019-3087

Sadjadi, S. O., & Hansen, J. H. L. (2014). Blind spectral weighting for robust speaker identification under reverberation mismatch. *IEEE/ACM Transactions on Audio Speech and Language Processing*, *22*(5), 937–945. https://doi.org/10.1109/taslp.2014.2311329

Scikit-Learn. (2024). *User guide: Contents — scikit-learn 0.22.1 documentation*. Scikit-Learn.org. https://scikit-learn.org/stable/user_guide.html

Shen, J., Pang, R., Weiss, R., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Ryan, R. S., Saurous, R. A., Agiomvrgiannakis, Y., & Wu, Y. (2018). Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.4779-4783. https://doi.org/10.1109/ICASSP.2018.8461368

Snyder, D., GarciaRomero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). XVectors: Robust DNN embeddings for speaker recognition. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.5329-5333. https://doi.org/10.1109/ICASSP.2018.8461375

SoX Development Team. (2016). *Source code for sox*. GitHub. https://github.com/skratchdot/sox/blob/master/src

SoX Development Team. (2024). *SoX - sound eXchange*. SourceForge. https://sourceforge.net/projects/sox/

Stupp, C. (2019). *Fraudsters used AI to mimic ceo's voice in unusual cybercrime case*. Wall Street Journal. https://www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402

Tak, H., Patino, J., Todisco, M., Nautsch, A., Evans, N., & Larcher, A. (2021). EndtoEnd antispoofing with rawnet2. *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.6369-6373. https://doi.org/10.1109/ICASSP39728.2021.9414234

Todisco, M., Delgado, H., & Evans, N. (2017). Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification. *Computer Speech & Language*, *45*, 516–535. https://doi.org/10.1016/j.csl.2017.01.001

Todisco, M., Wang, X., Vestman, V., Sahidullah, M., Delgado, H., Nautsch, A., Yamagishi, J., Evans, N., Kinnunen, T., & Lee, K. A. (2019). ASVspoof 2019: Future horizons in spoofed and fake audio detection. *Arxiv*. https://doi.org/10.48550/arXiv.1904.05441

Wang, Y., Ryan, R. S., Stanton, D., Wu, Y., Weiss, R., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q. V., Agiomyrgiannakis, Y., Clark, R., & Saurous, R. A. (2017). Tacotron: Towards end-to-end speech synthesis. *Arxiv*. https://doi.org/10.21437/interspeech.2017-1452

Xue, J., Fan, C., Lv, Z., Tao, J., Yi, J., Zheng, C., Wen, Z., Yuan, M., & Shao, S. (2025). Audio deepfake detection based on a combination of F0 information and real plus imaginary spectrogram features. *Association for Computing Machinery*, pp.19-26. https://doi.org/10.1145/3552466.3556526

Yamagishi, J., Todisco, M., Sahidullah, M., Delgado, H., Wang, X., Evans, N., Kinnunen, T., Lee, K. A., Vestman, V., & Nautsch, A. (2019). Asvspoof 2019: Automatic speaker verification spoofing and countermeasures challenge evaluation plan. *ASV Spoof*, *13*.

Yamagishi, J., Wang, X., Todisco, M., Sahidullah, M., Patino, J., Nautsch, A., Liu, X., Lee, K. A., Kinnunen, T., Evans, N., & Delgado, H. (2021). ASVspoof 2021: Accelerating progress in spoofed and deepfake speech detection. *Arxiv*. https://doi.org/10.48550/arXiv.2109.00537

Zhou, X., Garcia-Romero, D., Duraiswami, R., Espy-Wilson, C., & Shamma, S. (2011). Linear versus mel frequency cepstral coefficients for speaker recognition. *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*. https://doi.org/10.1109/asru.2011.6163888

Zhu, Q.-S., Zhang, J., Zhang, Z., Wu, M., Fang, X., & Dai, L.-R. (2022). A noise-robust self-supervised pre-training model based speech representation learning for automatic speech recognition. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.3174-3178. https://doi.org/10.1109/icassp43922.2022.9747379

APPENDIX A

This appendix contains information for what the waveforms and the frequency spectra of impulse responses (IRs)look like before and after the moving average smoothing techniques. The shapes and lines in blue represent the IRs before smoothing, while those in orange represent the IRs after smoothing. Three window sizes (size of 5 bins, 15 bins, 30 bins) were applied for the moving average techniques to experiment with different levels of smoothness. The figures below are examples from one IR, displaying from small windows to larger windows.

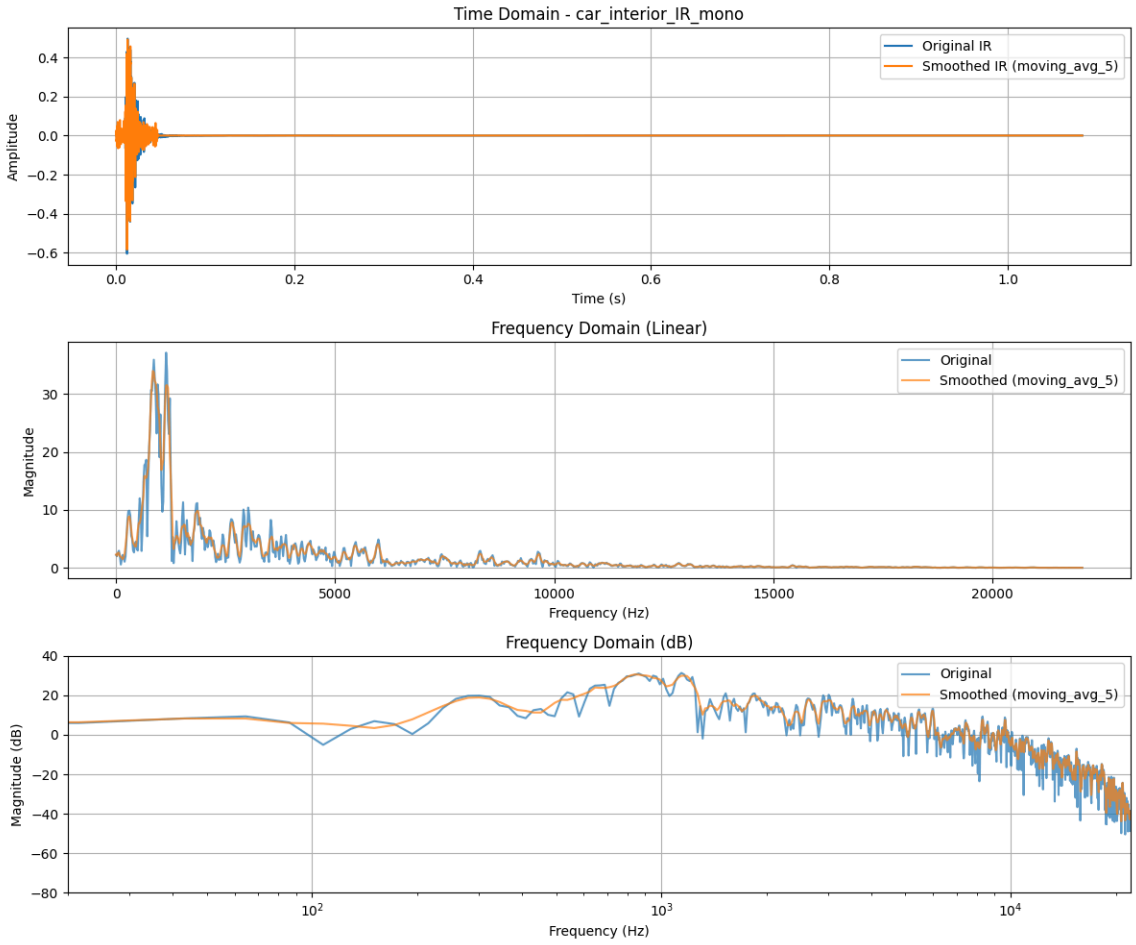

Figure 39: Original vs. Smoothed Car Interior IR (Window Size = 5)

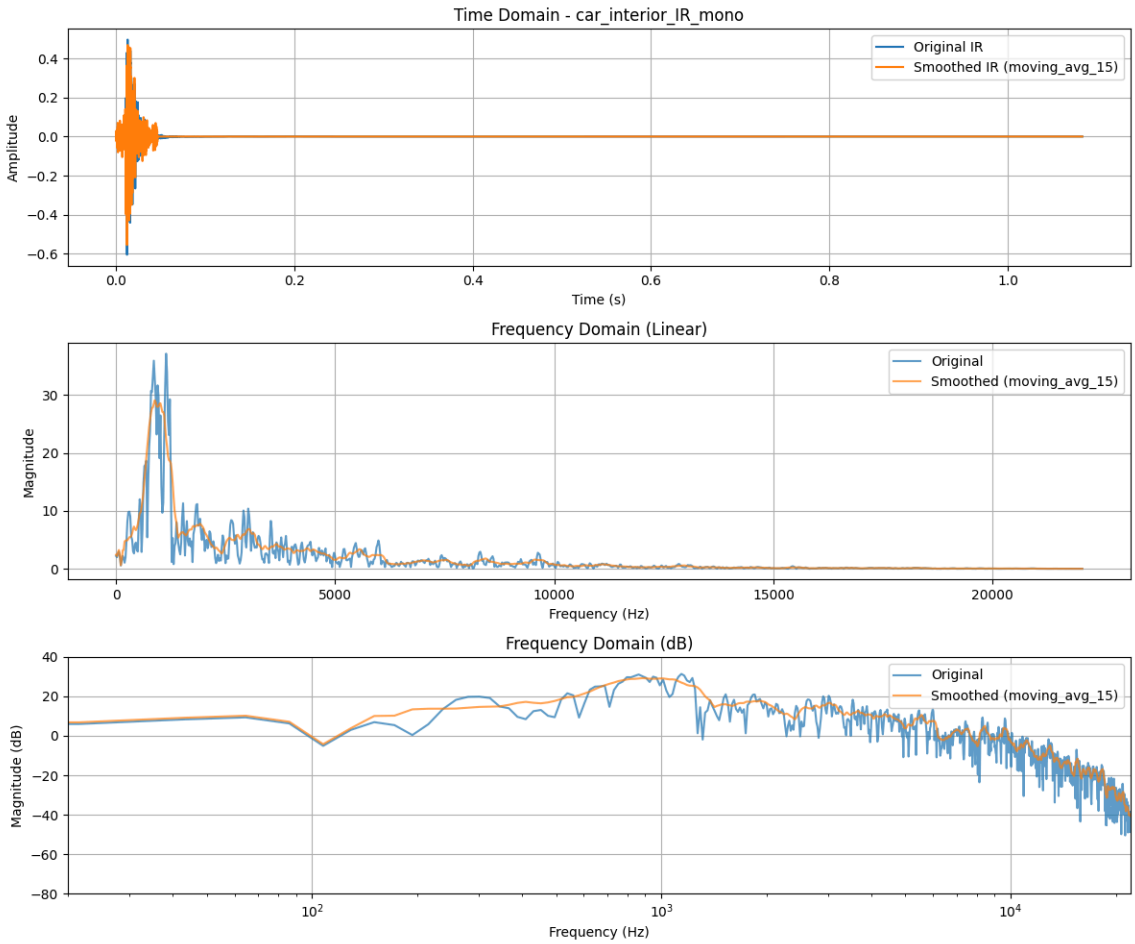

Figure 40: Original vs. Smoothed Car Interior IR (Window Size = 15)

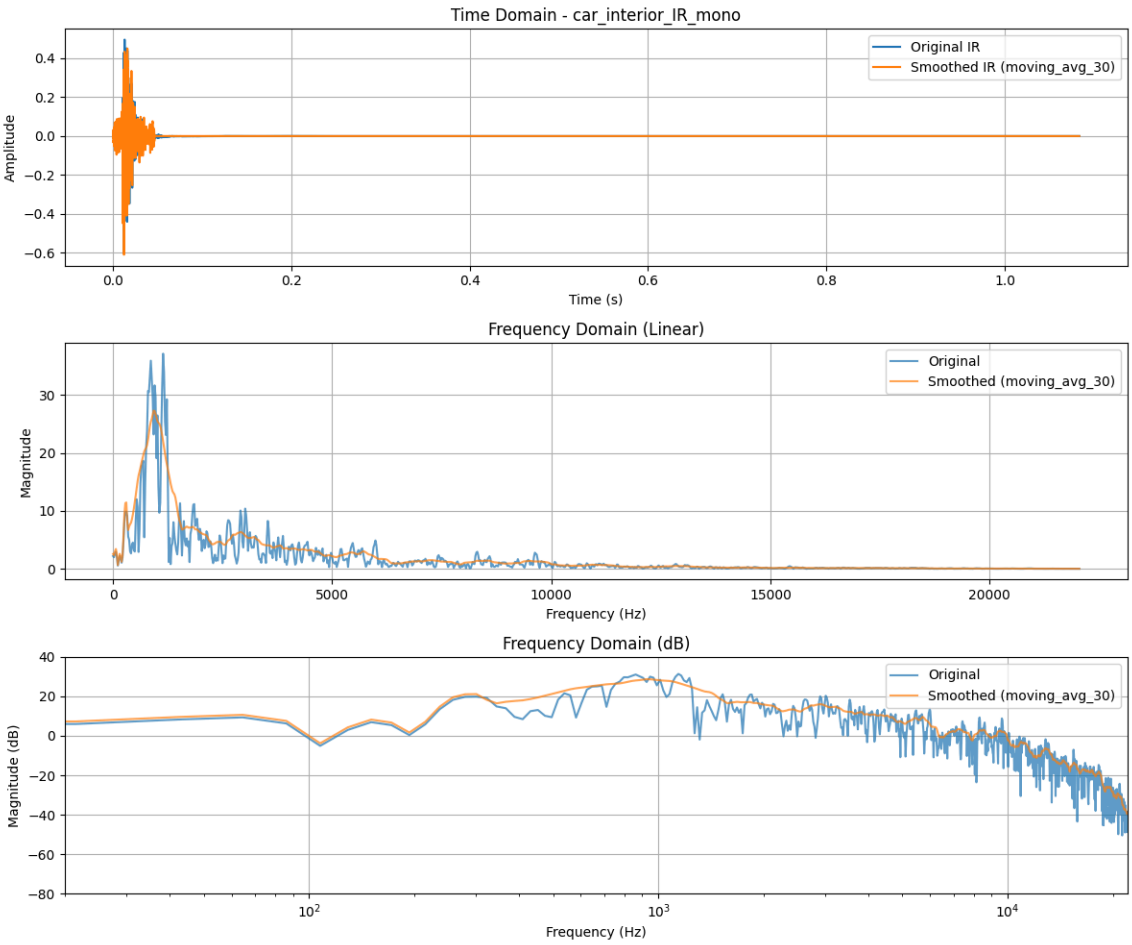

Figure 41: Original vs. Smoothed Car Interior IR (Window Size = 30)