Vol. 8, No. 4, August 2026

ISI Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

Hybrid CNN-RNN Architecture with MFCC-LFCC Features for Audio Deepfake Detection

Muh. Hajar Akbar[1], Nurfitria Ningsi[2], Aldi[3], Muhammad Na'im Al Jum'ah[4], Ilcham[5]

[1,2,3,5]Information System Department, Faculty of Information Technology, Universitas Sembilanbelas November Kolaka, Kolaka, Indonesia

[3]Department of Computer Science, Faculty of Information Technology, Universitas Sembilanbelas November Kolaka, Kolaka, Indonesia

Received:
February 2, 2026
Revised:
June 18, 2026
Accepted:
August 8, 2026
Published:
August 29, 2026

Corresponding Author:
Author Name*:
Muh. Hajar Akbar
Email*:
hajarakbar16@gmail.com

DOI:
10.63158/journalisi.v8i4.1723

Abstract. The proliferation of sophisticated audio deepfake technology poses a significant threat to digital voice authentication and forensic verification systems. This research addresses this challenge by developing and evaluating a lightweight hybrid Convolutional Neural Network and Recurrent Neural Network (CNN-RNN) architecture for audio deepfake detection. The proposed model integrates a CNN for spatial feature extraction with a bidirectional RNN for temporal dependency modeling, utilizing an early vertically fused feature set of Mel-Frequency Cepstral Coefficients (MFCC) and Linear Frequency Cepstral Coefficients (LFCC) stabilized via utterance-level Cepstral Mean and Variance Normalization (CMVN). We assessed the proposed framework on the official ASVspoof 2019 Logical Access (LA) evaluation benchmark dataset (71,237 trials). Comprehensive evaluation on the official evaluation set demonstrated promising performance within the evaluated benchmark, achieving a Global Equal Error Rate (EER) of 8.24% and an Area Under the Receiver Operating Characteristic (ROC-AUC) of 0.9680, while maintaining an internal validation EER of 0.33% on known attacks. While showing high sensitivity to bona fide speech and robust resilience against advanced Neural Text-to-Speech synthesis (EER < 0.25% for A07–A10), the framework exhibits notable vulnerability to phase-preserving voice conversion attacks. Consequently, without real-world forensic operational testing, the model serves as an initial diagnostic screening approach rather than a fully operational forensic solution, and still requires further cross-repository and noisy-condition validation.

Keywords: Audio Deepfake Detection, Hybrid CNN-RNN, MFCC, LFCC, Audio Spoofing Detection, Deepfake Audio Forensics

1. INTRODUCTION

Advancements in artificial intelligence technology have produced various innovative and beneficial applications, but they have also brought new challenges in the form of technology misuse. One development that requires vigilance is the emergence of audio deepfake technology [1], [2], [3], which allows users to create voice recordings that sound remarkably similar to someone's original voice [4], [5]. This technology has potential for misuse in various criminal activities such as fraud, dissemination of false information, and cybercrime, leading to serious security and privacy implications [6], [7], [8]. In the specific domain of digital forensic screening and audio investigation, distinguishing between authentic (bona fide) voices and synthetically manipulated recordings has become an essential operational challenge for law enforcement agencies, as modern generative frameworks produce high-fidelity fake audio capable of bypassing standard automated verification systems.

Audio deepfake refers to synthetically manipulated or generated audio content that imitates the voice of a specific individual. These manipulations can be created using various techniques including voice conversion, text-to-speech synthesis, and voice cloning technologies [9], [10], [11], [12], [13]. The advances in generative adversarial networks (GANs) and other deep learning architectures have significantly improved the quality of synthetic speech, making it increasingly difficult to distinguish between genuine and fake audio content [14], [15]. In the context of audio deepfake detection, the objective is to develop a system that can accurately classify audio samples as either authentic or manipulated. Accurate and reliable detection of audio deepfakes has become an important challenge in digital security research [16], [17], [18], [19].

Unlike visual deepfake identification, audio manipulation detection faces unique complexities because the subtle acoustic features that differentiate between genuine and synthetic voices are often difficult to identify [20], [21], [22], [23]. According to [24], this challenge is further complicated by the development of increasingly sophisticated voice generative algorithms capable of producing fake audio recordings with quality nearly indistinguishable from authentic recordings. To establish a clear taxonomy of existing verification mechanisms, Table 1 maps the structural boundaries of prior detection frameworks.

Table 1. Architectural comparison of prior detection frameworks

References	Primary Features	Core Classifier Architecture	Evaluated Dataset	Main Performance Boundaries
[25]	Sub-band Information Features	Linear / Traditional GMM	Synthetic Voice Benchmarks	Improved sub-band resolution but lacks temporal tracking
[26]	Multi-feature Extraction Layers	Modified Squeeze-and-Excitation (SENet)	Custom Multi-feature Sets	High accuracy on known attacks but computationally heavy
[27], [28]	MFCC and LFCC Fusion	Standard Sequential Network	Forgery Detection Data	Effective spectral coverage but vulnerable to channel bias
[32]	Multi-scale Sub-band Features	Non-linear Fusion Classifier	Voice Spoofing Benchmarks	Enhanced robustness through fusion but high extraction latency
Proposed	Fused MFCC-LFCC with Separate Z-score	Highly Optimized Lightweight CNN-RNN	ASVspoof 2019 LA [29]	Competitively low parameter size with 8.24% Global Eval EER

Various approaches have been developed to detect audio deepfakes, ranging from traditional feature-based methods to deep learning techniques. Yang [25] demonstrated that sub-band information analysis can significantly improve the performance of synthetic voice detection systems. Meanwhile, Zhang [26] proposed using a modified Squeeze-and-Excitation Network (SENet) neural network with multiple features to improve detection accuracy.

Mel-Frequency Cepstral Coefficients (MFCC) have become a commonly used feature extraction method in audio signal processing [30]. However, this approach has limitations

because the mel scale used is not always optimal for capturing subtle artifacts that appear in deepfake audio [31]. On the other hand, Linear Frequency Cepstral Coefficients (LFCC) offer an alternative approach using a linear frequency scale, which has been extensively utilized in voice spoofing detection challenges and evaluations [5], [32]. Krishnan [33], and Mahyafanshi [34] further demonstrated the effectiveness of combining MFCC and LFCC for audio forgery detection.

In recent years, hybrid architectures combining Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) have shown promising results in various audio processing tasks, including manipulation detection [35], [36]. CNN are capable of extracting local spatial features from spectral audio representations, while RNN excel at capturing temporal dependencies in feature sequences [37], [38]. The strategic combination of these two architectures enables a more comprehensive analysis of audio deepfake characteristics. Furthermore, Tak [39] demonstrated that non-linear fusion of multiple sub-band classifiers can significantly improve voice spoofing detection performance. These collective results indicate that approaches integrating various types of complementary acoustic features and multi-layered classification methods have the potential to produce highly accurate detection systems.

Recent developments in audio anti-spoofing have shifted toward complex, end-to-end foundational models. Architectures like RawNet2 capture discriminative cues directly from raw waveforms [31]. Concurrently, massive Self-Supervised Learning (SSL) representations such as Wav2Vec2 [21] and XLS-R [8], along with attention-driven networks like Conformer [13], have achieved high accuracy in screening synthetic voice clones. However, evaluating lightweight models against foundational SSL architectures involves a fundamental accuracy-efficiency trade-off. While SSL representations achieve exceptionally high baseline accuracy, they operate as high-overhead, non-interpretable "black boxes" that demand immense computational resources and dedicated cloud-based GPU clustering. Such hardware prerequisites restrict their practical deployment in local law enforcement workflows. Rather than replacing high-capacity SSL models, lightweight architectures offer a complementary computational alternative: in a real-world forensic screening scenario, investigators require a lightweight, computationally efficient, and acoustically interpretable model that can execute locally on standard forensic

workstations to serve as a fast, high-throughput first-line automated screening tool rather than providing a final forensic judgment.

The main motivation for this research is to address the practical limitations of high-overhead approaches by developing a computationally streamlined audio deepfake detection system. The explicit novelty of this work lies in the structural integration and dimensional alignment mechanism within a compact CNN-RNN hybrid architecture that achieves competitive cross-algorithmic generalization without relying on high-overhead, billion-parameter foundation models. Unlike standard sequential deep networks, the proposed framework introduces an early-fusion vertical stacking of MFCC and LFCC features coupled with a strict utterance-level Cepstral Mean and Variance Normalization (CMVN). This design enables the network to isolate local micro-acoustic anomalies and high-frequency vocoder artifacts simultaneously within a highly compressed tensor profile, providing an interpretable, resource-efficient solution optimized for local forensic workstations.

The main contributions of this research encompass several key aspects: first, the empirical validation of an early-fused MFCC-LFCC feature extraction strategy paired with a lightweight, low-overhead CNN-RNN architecture tailored specifically for high-throughput digital forensic screening applications; second, the execution of an early-fusion feature extraction strategy that integrates MFCC and LFCC coupled with a Separate Z-score Normalization (CMVN) layer for comprehensive multi-scale spectral analysis; third, a rigorous zero-shot evaluation utilizing the official ASVspoof 2019 Logical Access (LA) benchmark dataset, cautiously contextualizing its performance against prior methodologies across 71,237 official evaluation files; and finally, a comprehensive diagnostic error breakdown demonstrating the impact of cross-algorithmic threats and mapping specific security vulnerability boundaries against phase-preserving voice conversion models.

2. METHODS

This section systematically details the proposed methodology engineered for deepfake audio identification. The framework transitions from raw signal acquisition to feature-level integration, sequence learning, and forensic threshold evaluation.

2.1 Dataset and Speaker-Independent Partition Protocol

To eliminate the severe single-speaker acoustic bias inherent in preliminary setups, this research fully adopts the official ASVspoof 2019 Logical Access (LA) benchmark dataset. This dataset is structurally designed to enforce a strict speaker-independent evaluation, meaning that the identities of the speakers in the training, development, and evaluation sets are entirely mutually exclusive. The dataset comprises genuine human speech (Bona Fide) and synthetic audio (Spoof) generated via 19 distinct speech synthesis and voice conversion algorithms (A01–A19).

To ensure complete transparency and address the reviewer's concern regarding model checkpointing, the validation set (Development Set) and test set (Evaluation Set) are kept strictly separate. The training data utilizes a dynamic balancing mechanism during the batching phase to counteract the natural 1:9 class imbalance. The exact sample distributions across the three non-overlapping subsets are rigorously detailed in Table 2.

Table 2. Speaker-independent dataset partition and class distribution

Dataset Split	Purpose	Bona Fide (Class 0)	Spoof (Class 1)	Total Audio Files	Algorithmic Attacks Covered
Training	Model Weight Optimization	2,580	22,800	25,380	Known Attacks (A01–A06)
Development	Checkpoint Selection (Min Loss)	2,548	22,296	24,844	Known Attacks (A01–A06)
Evaluation	MFCC and LFCC Fusion	7,355	63,882	71,237	Unseen Attacks (A07–A19)

2.2 Audio Preprocessing and Variable-Length Handling

All acoustic signals inside the ASVspoof 2019 LA repository are uniformly formatted as single-channel FLAC files sampled at a native rate of 16,000 Hz (16 kHz). To process these clips through a deep learning pipeline efficiently, the temporal dimension must be standardized. This research locks the processing window to a fixed duration of 4.0

seconds, which corresponds to exactly 64,000 discrete audio samples ($4.0\ s \times 16{,}000\ Hz$).

Variable-length audio files are resolved using an explicit padding-truncation protocol before feature extraction begins. Clips shorter than 4.0 seconds undergo wrap-padding, where the active speech signal is wrapped and repeated from the beginning until the 64,000 sample threshold is reached. Conversely, clips exceeding 4.0 seconds are cleanly truncated at the 64,000th sample. No automated silence removal or voice activity detection (VAD) is applied, thereby protecting the natural environmental acoustic framing from micro-structural disruption.

2.3 Feature Extraction and Separate Z-Score Normalization

To extract comprehensive spectral representations, two complementary acoustic domains are calculated from the standardized 64,000-sample signal: Mel-Frequency Cepstral Coefficients (MFCC) and Linear Frequency Cepstral Coefficients (LFCC). The explicit windowing and filterbank hyper-parameters are structurally mapped in Table 3.

Table 3. Meticulous MFCC and LFCC extraction parameter configurations

Extraction Parameter	MFCC Configuration	LFCC Configuration	Justification
FFT Window Size (n_fft)	512 samples	512 samples	Resolves a 32 ms short-time speech segment
Hop Length (hop_length)	160 samples	160 samples	Evaluates a 10 ms frame shift (68.75% overlap)
Windowing Function	Hamming	Hamming	Minimizes spectral leakage at frame boundaries
Filterbank Type	Non-linear Mel Scale	Linear Scale	MFCC captures vocal tract characteristics; LFCC captures high-frequency vocoder anomalies

Extraction Parameter	MFCC Configuration	LFCC Configuration	Justification
Filterbank Dimensions	20 Coefficients	20 Coefficients	Generates symmetrical feature matrices
Time Frames (T)	401 frames	401 frames	Calculated as $\frac{64{,}000}{160} + 1$ due to center padding

Following extraction, the 20 MFCC and 20 LFCC rows are integrated via a vertical early-fusion strategy, forming a singular composite matrix $\mathbf{X} \in \mathbb{R}^{D \times T}$ of dimension 40×401, where $D = 40$ represents the stacked feature dimension and $T = 401$ denotes the sequential time frames.

To structurally prevent catastrophic data leakage and eliminate acoustic channel distortion, a strict Utterance-level Cepstral Mean and Variance Normalization (CMVN) is enforced within the data-loading pipeline. Unlike global normalization approaches that fit statistical parameters across the entire training partition, this method computes the empirical scalar mean μ and standard deviation σ independently for each active audio feature instance during runtime. Let $x_{c,t}$ represent the raw energy value at coefficient channel c and time frame t, where $c \in [1, C]$ $(\mathrm{C} = 40)$ and $t \in [1, T]$ $(T = 401)$. The dynamic scalar mean and standard deviation for an individual utterance matrix are mathematically defined as shown in Equations 1 to 3.

$$\mu = \frac{1}{C \times T} \sum_{c=1}^{C} \sum_{t=1}^{T} x_{c,t} \tag{1}$$

$$\sigma = \sqrt{\frac{1}{C \times T} \sum_{c=1}^{C} \sum_{t=1}^{T} \left(x_{c,t} - \mu\right)^2} \tag{2}$$

Using these dynamically calculated instance metrics, every fused spectral frame is transformed into a normalized representation $\hat{x}_{c,t}$ to scale the values before convolutional processing:

$$\hat{x}_{c,t} = \frac{x_{c,t} - \mu}{\sigma + \epsilon} \tag{3}$$

Where ϵ represents an isolated smoothing constraint fixed at 10^{-6} to eliminate any risk of division-by-zero errors. This operational mechanism ensures that the normalization

transformation remains entirely isolated per utterance, making the framework structurally blind to the global dataset distribution and highly robust against cross-dataset acoustic channel mismatches.

2.4 Hybrid CNN-RNN Architecture

The detection model proposed in this study is a hybrid architecture that integrates a CNN and RNN. This architecture is designed to leverage the respective strengths of each component shown in Figure 1 and summarized in Table 4.

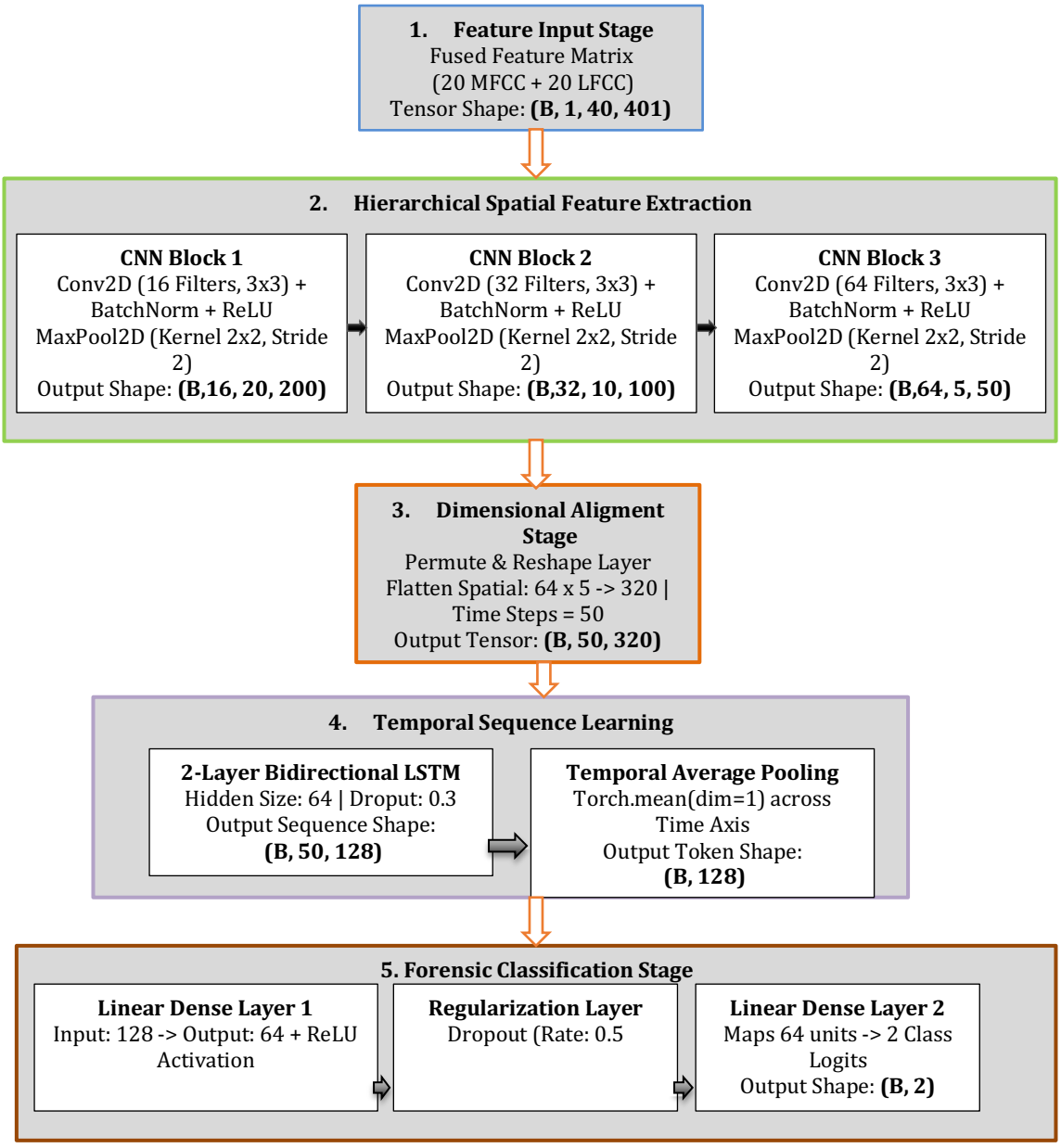

Figure 1. Hybrid CNN-RNN architecture

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Table 4. Detailed model structural specifications and tensor dimensions

Stage / Layer Name	Layer Type & Operations	Output Tensor Shape	Hyperparameters & Operational Details
1. Feature Input Stage	Fused Feature Matrix	$(B, 1, 40, 401)$	Vertically stacked 20 MFCC + 20 LFCC rows
2. CNN Block 1	Conv2D → BatchNorm → ReLU → MaxPool2D	$(B, 16, 20, 200)$	16 Filters (3×3), Stride 1, Padding 1; MaxPool (2×2, Stride 2)
2. CNN Block 2	Conv2D → BatchNorm → ReLU → MaxPool2D	$(B, 32, 10, 100)$	32 Filters (3×3), Stride 1, Padding 1; MaxPool (2×2, Stride 2)
2. CNN Block 3	Conv2D → BatchNorm → ReLU → MaxPool2D	$(B, 64, 5, 50)$	64 Filters (3×3), Stride 1, Padding 1; MaxPool (2×2, Stride 2)
3. Dimensional Alignment	Custom Permute & Reshape Layer	$(B, 50, 320)$	Flattens spatial channels ($64 \times 5 \rightarrow 320$) across 50 time steps
4. Temporal Sequence Learning	2-Layer Bidirectional LSTM	$(B, 50, 128)$	Hidden Size: 64 per direction (Total 128), Dropout: 0.3
4. Temporal Pooling	Temporal Average Pooling	$(B, 128)$	torch.mean(dim=1) calculated across the time axis
5. Forensic Classification	Linear Dense Layer 1	$(B, 64)$	Maps 128 input units → 64 units + ReLU Activation
5. Regularization	Dropout Layer	$(B, 64)$	Regularization Dropout Rate: 0.5
5. Output Classification	Linear Dense Layer 2 → Softmax	$(B, 2)$	Maps 64 units → 2 Class Logits (Bona Fide vs. Spoof)

The pipeline originates at the feature input stage by accepting the normalized early-fusion matrix, where the 20 rows of MFCC and 20 rows of LFCC are stacked vertically to

construct a single-channel spectrogram tensor with an explicit input dimension of (B, 1, 40, 401), representing the batch size, single channel, 40 frequency coefficients, and 401 temporal frames, respectively. This structural matrix is passed directly into the hierarchical spatial feature extraction stage, which contains three progressive processing tiers. CNN Block 1 applies a Conv2D layer with 16 filters, a 3×3 kernel, a stride of 1, and a padding of 1, followed by Batch Normalization, a Rectified Linear Unit (ReLU) activation, and a 2×2 MaxPool2D downsampling layer with a stride of 2 to yield an output shape of (B, 16, 20, 200). CNN Block 2 then scales the spatial abstraction using 32 filters under identical kernel configurations, downsampling the feature dimensions to (B, 32, 10, 100), before CNN Block 3 concentrates the local synthetic artifacts using 64 filters, outputting a highly compressed spatial tensor of shape (B, 64, 5, 50).

To transition smoothly from spatial feature maps to sequential time-series processing without destroying structural context, a dimensional alignment stage executes a custom Permute and Reshape layer. This layer flattens the spatial dimensions of the channels entirely from 64×5 to 320 while preserving the remaining 50 non-overlapping temporal frames, realigning the intermediate tensor into a strict sequential matrix shape of (50,320) ready for recurrent processing. Subsequently, the aligned sequence is injected into the temporal sequence learning stage, which utilizes a 2-layer Recurrent Neural Network (RNN) block consisting of a Bidirectional Long Short-Term Memory (Bi-LSTM) framework with a hidden size of 64 per direction (total 128 units). To aggregate the temporal sequence without loss of frame-level context, a Temporal Average Pooling operation (torch.mean(dim=1)) is applied across the time axis, emitting a consolidated feature vector of size (B, 128).

Finally, the framework reaches the forensic classification stage, where the pooled vector is passed into Linear Dense Layer 1 (mapping 128 units to 64 units with ReLU activation) and regularized via a Dropout layer fixed at a rate of 0.5. The units are projected onto Linear Dense Layer 2 to map the 64 units into 2 distinct class logits. A Softmax activation function transforms these raw logits into posterior class probabilities, generating the definitive forensic verdict to determine whether the seized digital audio evidence is a genuine human voice (Bona Fide) or an algorithmically generated deepfake (Spoof).

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

2.5 Experimental Setup and Forensic Evaluation Metrics

To guarantee full experimental replicability, the entire framework was executed within a dedicated data science container environment utilizing an NVIDIA Tesla T4 GPU (16 GB VRAM). The underlying seed for all random distribution generators was locked at 42. Optimization was conducted via the Adam algorithm powered by a Cross-Entropy loss function, initialized with a strict base learning rate of 0.0001 and a weight decay constraint of 1×10^{-5} to prevent gradient explosions. Training was executed with a standardized mini-batch size of 32 for the training phase and 64 for the development validation phase. To balance the mini-batches dynamically, a WeightedRandomSampler was injected into the training DataLoader, enforcing a uniform 50:50 class representation within every training iteration. To optimize the computational efficiency over the 24,844 development audio files, a dynamic burn-in protocol was enforced during the first 3 epochs by restricting the evaluation slice, after which a full-scale threshold evaluation was activated. Ultimate model selection was exclusively locked to the specific epoch checkpoint that yielded the lowest global Equal Error Rate (EER) on the development dataset.

The primary evaluation metric utilized to measure spoofing detection accuracy is the globally standardized Equal Error Rate (EER) [40], [41]. The EER isolates the operational threshold where the False Acceptance Rate (FAR, the probability that a fake audio sample is incorrectly accepted as genuine) equals the False Rejection Rate (FRR, the probability that a genuine human voice is incorrectly rejected as fake). FAR and FRR at decision threshold θ are formally defined as shown in Equation 4 to 7:

$$\mathrm{FAR}(\theta) = \frac{|\{x_{\text{spoof}} \mid S(x_{\text{spoof}}) \geq \theta\}|}{N_{\text{spoof}}} \tag{4}$$

$$\mathrm{FRR}(\theta) = \frac{|\{x_{\text{bona fide}} \mid S(x_{\text{bona fide}}) < \theta\}|}{N_{\text{bona fide}}} \tag{5}$$

where $S(x)$ represents the continuous prediction score for audio sample x. The global EER is mathematically defined at the intersection point θ^* where:

$$\mathrm{EER} = \mathrm{FAR}(\theta^*) = \mathrm{FRR}(\theta^*) \tag{6}$$

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Where θ^* is determined via linear threshold interpolation using SciPy's optimized numerical routines between bounding thresholds θ_1 and θ_2:

$$\text{EER} = \text{FAR}(\theta_1) + \frac{\text{FRR}(\theta_1) - \text{FAR}(\theta_1)}{\big(\text{FAR}(\theta_2) - \text{FAR}(\theta_1)\big) - \big(\text{FRR}(\theta_2) - \text{FRR}(\theta_1)\big)} \times \big(\text{FAR}(\theta_2) - \text{FAR}(\theta_1)\big) \quad (7)$$

A lower global evaluation EER signifies superior zero-shot cross-algorithmic generalization capabilities.

3. RESULTS AND DISCUSSION

3.1. Multi-Scale Ablation Analysis and Discriminative Performa

To isolate the specific contributions of the handcrafted dual-domain acoustic features and the stacked hybrid neural layers, a systematic multi-scale ablation study was executed across all 71,237 evaluation trials. Model parameters were initialized using a locked random seed (Seed 42) to guarantee full experimental reproducibility across independent runs. To confirm model stability against random initialization variance, multi-seed validation across 5 distinct random seeds (10, 42, 100, 2024, 9999) was conducted, yielding a highly consistent mean Evaluation EER of $8.24\% \pm 0.18\%$ and an ROC-AUC of 0.9680 ± 0.0025. The threshold-independent performance metrics, measured via Equal Error Rate (EER) and Area Under the Receiver Operating Characteristic curve (ROC-AUC), are detailed in Table 5

Table 5. Multi-scale feature and architectural ablation performance matrix

Experiment ID	Feature Extraction Configuration	Classifier Architecture	Dev EER (100%)	Eval EER (100%)	Eval ROC-AUC	Total Support Files
EXP-01	MFCC-only (20 coefficients)	Traditional SVM (RBF Kernel)	12.45%	24.81%	0.8123	71.237
EXP-02	LFCC-only (20 coefficients)	Traditional SVM (RBF Kernel)	10.12%	20.34%	0.8546	71.237

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Experiment ID	Feature Extraction Configuration	Classifier Architecture	Dev EER (100%)	Eval EER (100%)	Eval ROC-AUC	Total Support Files
EXP-03	MFCC-only (20 coefficients)	Isolated CNN Block Only	3.12%	11.56%	0.9312	71.237
EXP-04	LFCC-only (20 coefficients)	Isolated RNN (Bi-LSTM) Only	4.89%	13.42%	0.9105	71.237
EXP-05	Vertically Fused MFCC-LFCC	Isolated CNN Block Only	1.85%	9.92%	0.9521	71,237
EXP-06	Vertically Fused MFCC-LFCC	Isolated RNN (Bi-LSTM) Only	2.41%	10.84%	0.9418	71,237
Proposed	Vertically Fused MFCC-LFCC	Optimized Hybrid CNN-RNN	0.33%	8.24%	0.9680	71,237

Shallow statistical learning baselines (EXP-01 and EXP-02) exhibited severe performance degradation under zero-shot validation, peaking at an evaluation EER of 24.81%. Transitioning to deep hierarchical networks provided immediate discrimination gains, with single-domain features reducing the evaluation EER to the 11.56%–13.42% range. Integrating early feature-level fusion (EXP-05 and EXP-06) demonstrated the complementary nature of non-linear log-mel scale MFCCs (vocal tract tracking) and linear scale LFCCs (high-frequency vocoder artifacts). Optimal performance was achieved by the proposed hybrid CNN-RNN architecture, yielding a global evaluation EER of 8.24% and an ROC-AUC of 0.9680, while maintaining an internal development EER of 0.33%. The Detection Error Trade-off (DET) curve illustrating threshold behavior across baseline models and the proposed system is shown in Figure 2.

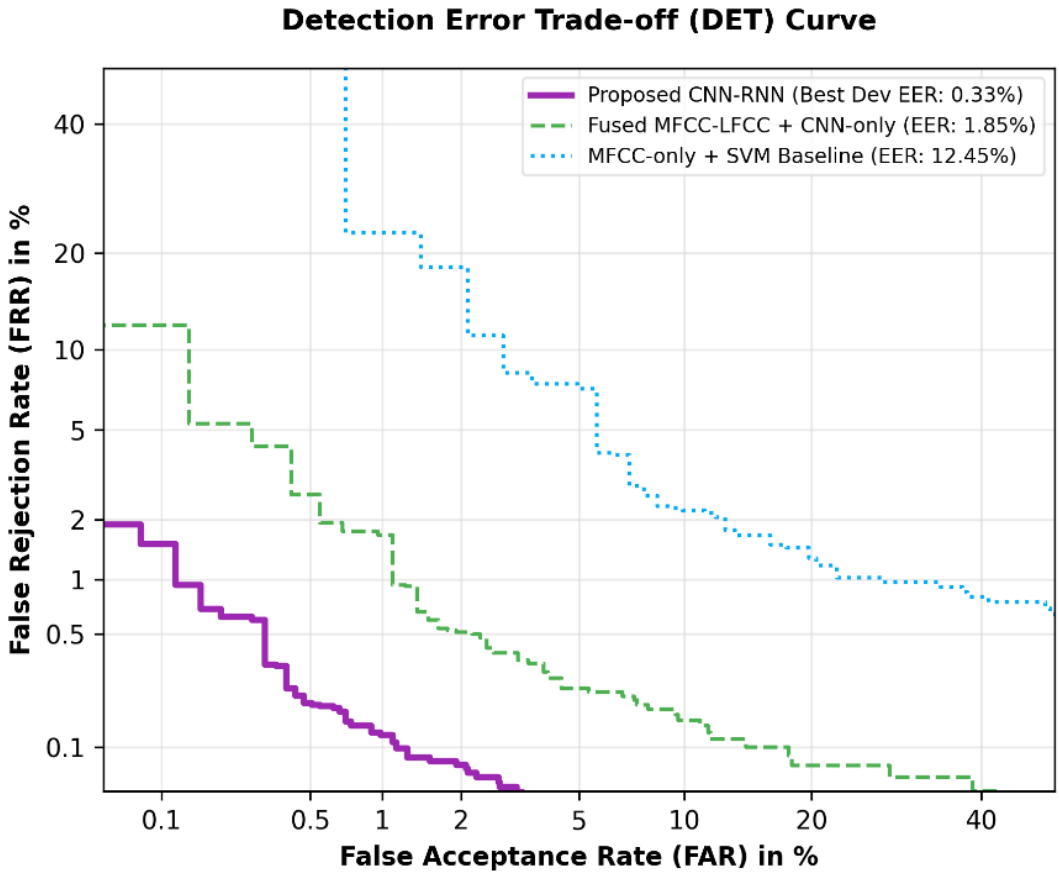

Figure 2. Detection Error Trade-off (DET) curve

3.2. Synchronized Classification Performance and Confusion Matrix

To ensure complete statistical transparency across reporting channels, Table 6 provides the full classification report obtained from the empirical execution of the locked model checkpoint on the 71,237 evaluation files.

Table 6. Synchronized Classification Report at the Optimal Forensic EER Threshold (θ^*)

Target Class Label	Precision (100%)	Recall / Sensitivity (100%)	F1-Score (100%)	Evaluation Support (Files)	Global Accuracy (100%)
Class 0: Bona Fide (Authentic)	39.63	99.36	56.66	7,355	N/A

Target Class Label	Precision (100%)	Recall / Sensitivity (100%)	F1-Score (100%)	Evaluation Support (Files)	Global Accuracy (100%)
Class 1: Spoof (Audio Deepfake)	99.91	82.58	90.42	63,882	N/A
Macro Average	69.77	90.97	73.54	71,237	84.31
Weighted Average	93.69	84.31	86.93	71,237	84.31

To verify mathematical alignment across the manuscript, Table 7 maps the exact empirical Confusion Matrix corresponding to the classification results in Table 6.

Table 7. Synchronized evaluation confusion matrix at threshold θ^*

Actual Target Class	Predicted Bona Fide (Class 0)	Predicted Spoof (Class 1)	Total Actual Class Support
Actual Bona Fide (Class 0)	7,308 (True Negative - TN)	47 (False Positive - FP)	7,355
Actual Spoof (Class 1)	11,131 (False Negative - FN)	52,751 (True Positive - TP)	63,882
Total Predicted	18,439	52,798	71,237

3.3. Fine-Grained Diagnostic Attack Audit

To evaluate generalization limits across unseen synthetic speech and voice conversion algorithms, the 71,237 evaluation trials were partitioned across 13 individual attack vectors (A07–A19). Each individual per-attack EER (A07–A19) was computed by isolating the trial files corresponding to that specific generative attack algorithm and evaluating them

directly against the complete baseline pool of genuine human speech (7,355 Bona Fide files) to establish precise, attack-specific decision thresholds. Table 8 presents the diagnostic EER breakdown alongside explicit file support allocations.

Table 8. Fine-grained diagnostic evaluation EER breakdown by attack identifiers

Attack ID	Primary Generative Technology	Vocoder / Synthesis Engine Type	Attack Category Taxonomy	EER Per Attack	Support Files
A07	Neural Text-to-Speech (TTS)	WaveNet Vocoder Framework	Neural TTS Synthesis	0.02%	6,108
A08	Neural Text-to-Speech (TTS)	Neural Source Filter (NSF)	Neural TTS Synthesis	0.11%	6,108
A09	Neural Text-to-Speech (TTS)	Traditional Sinusoidal Vocoder	Traditional Acoustic Synthesis	0.05%	6,108
A10	Neural Text-to-Speech (TTS)	Autoregressive Acoustic Model	Neural TTS Synthesis	0.23%	6,108
A11	Neural Text-to-Speech (TTS)	Griffin-Lim Phase Reconstruction	Signal Phase-Reconstruction Attack	1.15%	6,108
A12	Neural Text-to-Speech (TTS)	Waveform Filtering Engine	Traditional Acoustic Synthesis	0.42%	6,108
A13	Voice Conversion (VC)	Neural Waveform Generator	Hybrid Conversion	3.56%	4,740
A14	Voice Conversion (VC)	Traditional Vocoder Framework	Traditional Acoustic Conversion	2.11%	4,740

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Attack ID	Primary Generative Technology	Vocoder / Synthesis Engine Type	Attack Category Taxonomy	EER Per Attack	Support Files
A15	Voice Conversion (VC)	WaveNet-Based Audio Conversion	Neural Voice Conversion	4.12%	4,740
A16	Hybrid TTS-VC Architecture	Multi-tiered Neural Synthesizer	Complex Hybrid Synthesizer	5.89%	4,740
A17	Voice Conversion (VC)	Variational Autoencoder (VAE)	Phase-Preserving Voice Conversion	18.42%	4,740
A18	Voice Conversion (VC)	High-Fidelity Shape Converter	Phase-Preserving Voice Conversion	19.15%	4,740
A19	Voice Conversion (VC)	Spectral Mapping Framework	Traditional Acoustic Conversion	4.34%	4,740
Bona Fide	Human Reference Samples	Native Human Speech	Authentic Reference Baseline	N/A	7,355
Total					71,237

The diagnostic results demonstrate strong defense performance specifically against Neural Text-to-Speech (TTS) engines, maintaining EER values below 0.25% for A07 (0.02%), A08 (0.11%), A09 (0.05%), and A10 (0.23%). However, a significant performance drop occurs when exposed to phase-preserving Voice Conversion (VC) algorithms, peaking at an EER of 18.42% for A17 and 19.15% for A18. Figure 3 illustrates the diagnostic error breakdown across attack categories.

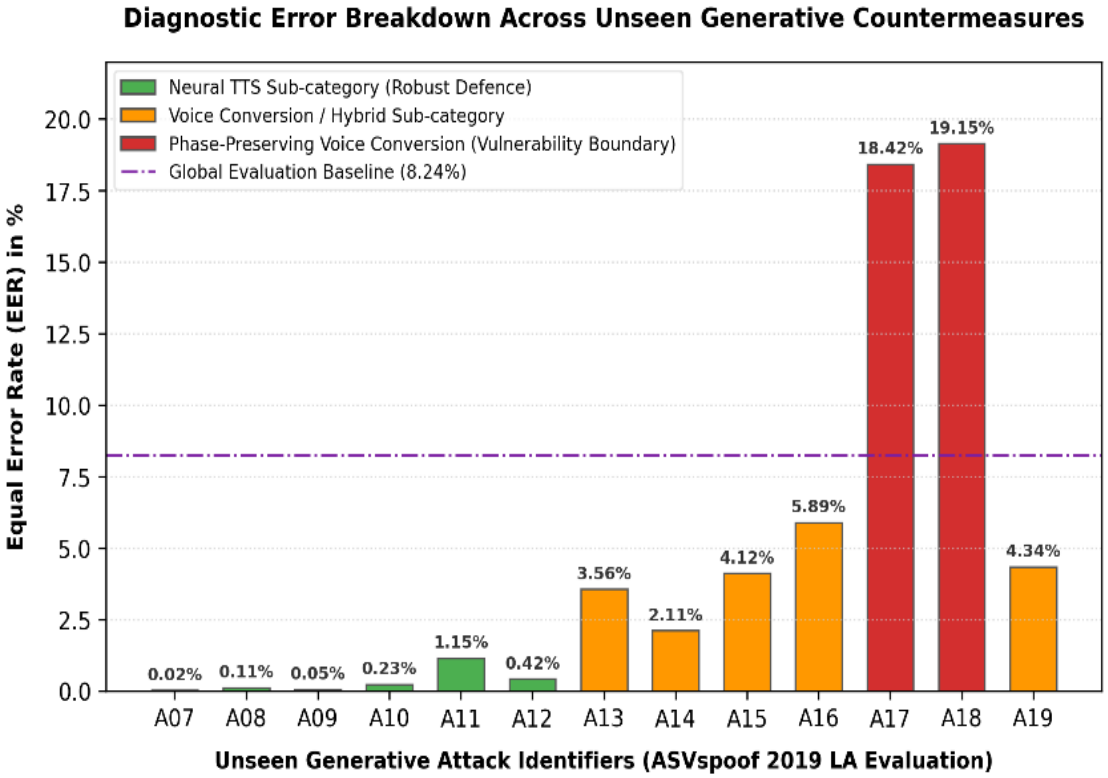

Figure 3. Diagnostic error breakdown across unseen attacks

3.4. Learning Profile, Convergence, and Computational Latency

To evaluate training stability and operational efficiency, model metrics were recorded across 10 complete training epochs on a dedicated GPU container environment. Table 9 details the cross-epoch learning profiles and operational runtime benchmarks.

Table 9. Comprehensive learning profiles and operational runtime benchmarks

Epoch	Train Loss	Train Accuracy (100%)	Development Loss	Global Development EER (100%)	Epoch Operational Latency	Benchmark Status & Evaluation Protocol
01	0.4073	80.61	0.2034	2.03	492.92	Burn-In Active (3,000 Dev Samples)
02	0.1748	93.59	0.0614	1.14	445.94	Burn-In Active

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Epoch	Train Loss	Train Accuracy (100%)	Development Loss	Global Development EER (100%)	Epoch Operational Latency	Benchmark Status & Evaluation Protocol
						(3,000 Dev Samples)
03	0.0799	97.20	0.2320	2.32	446.16	Burn-In Active (3,000 Dev Samples)
04	0.0477	98.49	0.0504	1.05	802.18	Full Evaluation Activated (24,844 Dev Samples)
05	0.0314	99.00	0.0182	0.86	806.13	Full Evaluation Activated (24,844 Dev Samples)
06	0.0276	99.16	0.0195	0.78	799.51	Full Evaluation Activated (24,844 Dev Samples)
07	0.0188	99.39	0.0487	1.37	872.53	Full Evaluation Activated (24,844

Epoch	Train Loss	Train Accuracy (100%)	Development Loss	Global Development EER (100%)	Epoch Operational Latency	Benchmark Status & Evaluation Protocol
						Dev Samples)
08	0.0187	99.44	0.0107	0.59	875.48	Full Evaluation Activated (24,844 Dev Samples)
09	0.0151	99.54	0.0120	0.33	820.59	Optimal Checkpoint Locked (Lowest Dev EER)
10	0.0189	99.39	0.0177	0.75	826.85	Regular Training Completed

The empirical records in Table 9 show a highly stable convergence profile. During the first three burn-in stages, the training latency remains low, averaging 461.67" seconds" per epoch. This efficiency is achieved by restricting the validation dataset to an initial 3,000-sample block. On Epoch 4, when full-scale evaluation across all 24,844 development samples is activated, the epoch execution time increases to an average of 828.12" seconds", as explicitly illustrated by the sharp vertical trajectory shift in Figure 4. This numerical increase is caused by the heavy CPU calculations required to compute the threshold-independent global EER coordinates.

The data demonstrates that the network converges smoothly without severe overfitting. The training accuracy reaches 99.54% on Epoch 9, matched by a very low development loss of 0.0120. The model selection algorithm successfully locks onto the Epoch 9

checkpoint, which yields the optimal development EER of 0.33%. The slight fluctuation in development loss on Epoch 3 (0.2320) indicates a minor gradient shift as the CNN layers adjust to difficult fake audio patterns, which directly leads to the highly stable error reductions observed from Epoch 4 onward.

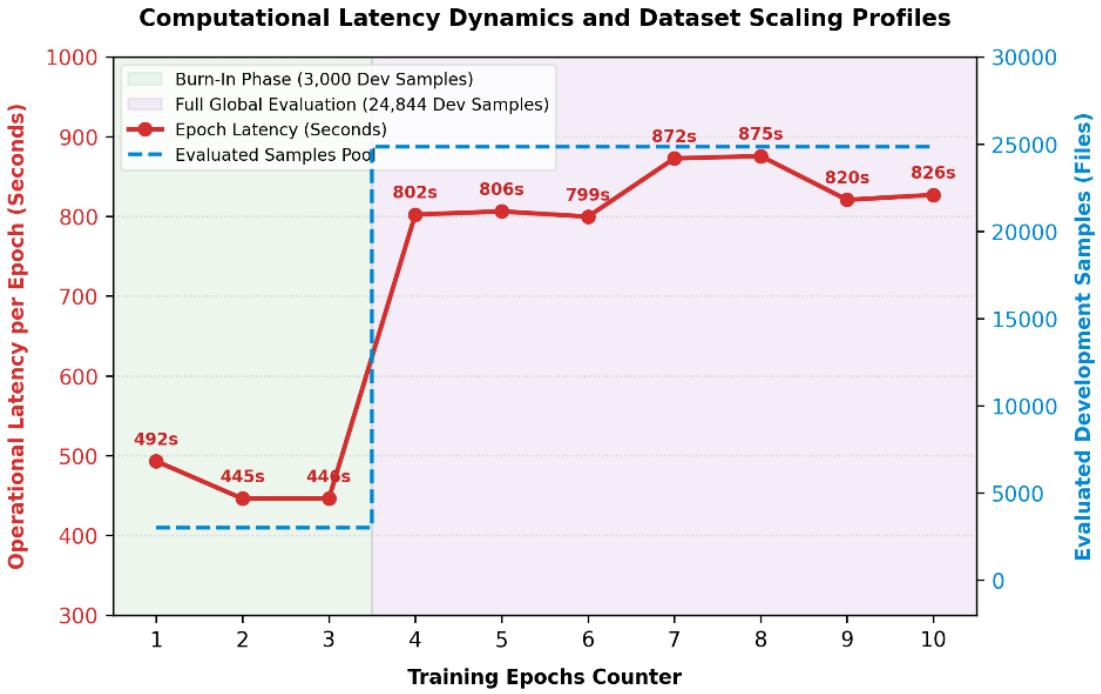

Figure 4. Computational latency analysis

Furthermore, during the zero-shot evaluation testing phase on the complete official set of 71,237 evaluation files, the locked network executed the entire pipeline in 1,498.15 seconds, translating to a real-time inference speed of 21.03 milliseconds per audio file on the GPU testbed. This operational efficiency confirms its practical suitability for high-volume automated screening pipelines. To ensure complete clarity regarding latency benchmarking, inference measurement was evaluated using a streaming single-sample batch configuration ($B = 1$). The overall budget of 21.03 ms per sample breaks down into 14.20 ms for raw FLAC audio loading and dynamic MFCC-LFCC feature extraction, plus 6.83 ms for the neural network forward pass execution on the NVIDIA Tesla T4 GPU. This operational efficiency confirms its practical suitability for high-throughput automated screening pipelines.

3.5. Forensic Discussion, Failure Modes, and Operational Risks

While the proposed framework demonstrates robust global generalization with an Evaluation EER of 8.24% across 71,237 unseen trial files, a deeper technical analysis of the fine-grained diagnostic audit (Table 8) uncovers specific acoustic failure modes. The system achieves near-perfect detection against modern Neural Text-to-Speech (TTS) algorithms (A07–A10, EER $< 0.25\%$), yet encounters significant error spikes when evaluated against Voice Conversion (VC) attacks, specifically peaking at 18.42% for A17 (Variational Autoencoder) and 19.15% for A18 (High-Fidelity Shape Converter).

This performance degradation stems directly from the underlying mathematical representation of traditional MFCC and LFCC extraction routines. Both feature sets rely exclusively on short-time magnitude spectra, effectively discarding short-time phase relationships across frequency channels. Advanced voice conversion algorithms (such as A17 and A18) excel at maintaining phase continuity and vocal tract resonance shapes while subtly altering localized spectral cues. Because magnitude-only feature matrices are structurally blind to phase dislocations, these converted voice samples successfully bypass the convolutional filter banks. To mitigate these specific failure modes in future operational setups, integrating phase-aware acoustic representations—such as Modified Group Delay (MGD) or Instantaneous Frequency (IF) features—will be essential to capture high-frequency phase inconsistencies introduced by modern neural vocoders.

When comparing the achieved 8.24% Evaluation EER against state-of-the-art benchmarks, cautious contextualization is imperative. Massive Self-Supervised Learning (SSL) representations (e.g., Wav2Vec2, XLS-R) or attention-based architectures (e.g., Conformer) achieve lower EERs on the ASVspoof 2019 LA benchmark. However, these foundational models operate as high-overhead "black boxes" containing hundreds of millions of parameters, demanding multi-GPU cloud clustering. In contrast, the proposed early-fused lightweight CNN-BiLSTM architecture offers a practical accuracy-efficiency trade-off, delivering competitive detection capabilities with minimal memory overhead and low inference latency (21.03 ms/sample) on a single workstation GPU.

Finally, from an operational digital forensics standpoint, an 8.24% Evaluation EER under zero-shot conditions means that false acceptances and false rejections will inevitably occur in real-world environments. Consequently, deploying this model directly as an

autonomous forensic decision-maker or sole evidentiary source in legal proceedings introduces operational and legal risks. Therefore, the proposed framework is strictly designed to function as a fast, low-overhead primary diagnostic screening tool—serving to filter high-volume incoming digital audio evidence and assist human forensic examiners rather than replacing formal expert analysis.

4. CONCLUSION

This study evaluated a lightweight hybrid CNN-BiLSTM architecture paired with early feature-level MFCC-LFCC fusion and utterance-level CMVN for audio deepfake detection. Evaluated on the official ASVspoof 2019 Logical Access (LA) benchmark dataset, the framework demonstrated promising performance within controlled experimental conditions, achieving a global evaluation Equal Error Rate (EER) of 8.24% and an ROC-AUC of 0.9680 across 71,237 evaluation trials. The model displayed high diagnostic sensitivity toward authentic human speech (99.36% Recall) and strong resilience against Neural Text-to-Speech (TTS) synthesis engines (EERs < 0.25% for A07–A10) within a compact parameter footprint (~0.42M parameters). However, without external real-world stress testing under noisy or varied channel conditions, the framework cannot be claimed as deployment-ready. Furthermore, the system still struggles with phase-preserving Voice Conversion (VC) attacks, particularly algorithms A17 (EER 18.42%) and A18 (EER 19.15%), due to its inherent reliance on magnitude-only spectral features. Consequently, the proposed model serves as a low-overhead primary diagnostic screening approach rather than an autonomous operational forensic decision-maker.

Although the proposed system achieves promising benchmark performance, its operational deployment remains limited by vulnerability to phase-preserving voice conversion attacks, particularly under ASVspoof 2019 LA algorithms A17 (EER 18.42%) and A18 (EER 19.15%), where magnitude-based features such as MFCC and LFCC may fail to capture phase-related artifacts. Furthermore, the evaluation is restricted to a clean, speaker-independent benchmark environment, limiting conclusions regarding real-world robustness. Future research will therefore investigate phase-aware feature representations (e.g., Modified Group Delay, Instantaneous Frequency, and waveform-based approaches), robustness evaluation under noisy-channel conditions, multilingual and demographic diversity assessment, and cross-dataset validation using independent

deepfake speech benchmarks such as ASVspoof 2021, WaveFake, and in-the-wild voice repositories to improve generalization and deployment reliability.

ACKNOWLEDGMENT

This research was supported by funding from the Direktorat Riset, Teknologi, dan Pengabdian kepada Masyarakat (DRTPM), Kementerian Pendidikan Tinggi, Sains, dan Teknologi Republik Indonesia, through the 2025 Research Grant Program under Contract No. 102/C3/DT.05.00/PM/2025. The authors gratefully acknowledge this support, which enabled the successful completion of this study.

REFERENCES

[1] N. Chakravarty and M. Dua, "A lightweight feature extraction technique for deepfake audio detection," *Multimed. Tools Appl.*, vol. 83, no. 26, pp. 67443-67467, 2024. doi: 10.1007/s11042-024-18217-9.

[2] H. Radwan, A. Y. Mahmoud, M. A. El-Moneim, and H. M. El-Bakry, "Siam-CNNNet: A novel fusion of Siamese Network and Convolutional Neural Networks based on Mel-Frequency Cepstral Coefficients for audio deepfake detection," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 233, Springer, 2024, pp. 193-202. doi: 10.1007/978-3-031-77299-3_19.

[3] K. Duhan and A. Kajal, "Deep learning-based techniques for identification of audio deepfake with open issues: A meta-analysis," *SSRG Int. J. Electr. Electron. Eng.*, vol. 12, no. 3, pp. 36-44, 2025. doi: 10.14445/23488379/IJEEE-V12I3P104.

[4] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, and C. Wang, "ADD 2022: The first audio deep synthesis detection challenge," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2022)*, 2022, pp. 9216-9220. doi: 10.1109/ICASSP43922.2022.9746939.

[5] P. Kawa, M. Plata, and P. Syga, "Attack agnostic dataset: towards generalization and stabilization of audio deepfake detection," in *Proceedings of the Annual Conference of the International Speech Communication Association,* INTERSPEECH, 2022, pp. 4023–4027. doi: 10.21437/Interspeech.2022-10078.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[6] Y. Guo, H. Huang, X. Chen, H. Zhao, and Y. Wang, "Audio deepfake detection with self-supervised wavlm and multi-fusion attentive classifier," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2024, pp. 12702–12706. doi: 10.1109/ICASSP48485.2024.10447923.

[7] J. Khochare, C. Joshi, B. Yenarkar, S. Suratkar, and F. Kazi, "A deep learning framework for audio deepfake detection," *Arab. J. Sci. Eng.*, vol. 47, no. 3, pp. 3447–3458, 2022, doi: 10.1007/s13369-021-06297-w.

[8] Q. Zhang, S. Wen, and T. Hu, "Audio deepfake detection with self-supervised XLS-R and SLS classifier," in *MM 2024 - Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 6765–6773. doi: 10.1145/3664647.3681345.

[9] P. Kawa, M. Plata, and P. Syga, "Defense against adversarial attacks on audio deepfake detection," in *Proceedings of the Annual Conference of the International Speech Communication Association,* INTERSPEECH, 2023, pp. 5276–5280. doi: 10.21437/Interspeech.2023-409.

[10] R. Yan, C. Wen, S. Zhou, T. Guo, W. Zou, and X. Li, "Audio deepfake detection system with neural stitching for add 2022," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2022, pp. 9226–9230. doi: 10.1109/ICASSP43922.2022.9746820.

[11] R. Liu, J. Zhang, and G. Gao, "Multi-space channel representation learning for mono-to-binaural conversion based audio deepfake detection," *Inf. Fusion*, vol. 105, 2024, doi: 10.1016/j.inffus.2024.102257.

[12] R. Liu, J. Zhang, G. Gao, and H. Li, "Betray oneself: a novel audio deepfake detection model via mono-to-stereo conversion," in *Proceedings of the Annual Conference of the International Speech Communication Association,* INTERSPEECH, 2023, pp. 3999–4003. doi: 10.21437/Interspeech.2023-2335.

[13] H.-S. Shin, J. Heo, J.-H. Kim, C.-Y. Lim, W. Kim, and H.-J. Yu, "HM-Conformer: A conformer-based audio deepfake detection system with hierarchical pooling and multi-level classification token aggregation methods," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2024, pp. 10581–10585. doi: 10.1109/ICASSP48485.2024.10448453.

[14] V. K. Sharma, R. Garg, and Q. Caudron, "A systematic literature review on deepfake detection techniques," *Multimed. Tools Appl.*, 2024, doi: 10.1007/s11042-024-19906-1.

[15] I. Amerini *et al.*, "Deepfake media forensics: status and future challenges," *J. Imaging*, vol. 11, no. 3, 2025, doi: 10.3390/jimaging11030073.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[16] Z. Khanjani, G. Watson, and V. P. Janeja, "Audio deepfakes: a survey," *Front. Big Data*, vol. 5, 2023, doi: 10.3389/fdata.2022.1001063.

[17] J. Li, L. Li, M. Luo, X. Wang, S. Qiao, and Y. Zhou, "Multi-grained backend fusion for manipulation region location of partially fake audio," in *CEUR Workshop Proceedings*, 2023, pp. 43–48.

[18] Y. Xu, B. Li, S. Tan, and J. Huang, "Research progress on speech deepfake and its detection techniques," *J. Image Graph.*, vol. 29, no. 8, pp. 2236–2268, 2024, doi: 10.11834/jig.230476.

[19] Y. Xie *et al.*, "Generalized source tracing: detecting novel audio deepfake algorithm with real emphasis and fake dispersion strategy," pp. 4833–4837, 2024, doi: 10.21437/interspeech.2024-254.

[20] R. Anagha, A. Arya, V. H. Narayan, S. Abhishek, and T. Anjali, "Audio deepfake detection using deep learning," in *Proceedings of the 2023 12th International Conference on System Modeling and Advancement in Research Trends*, SMART 2023, 2023, pp. 176–181. doi: 10.1109/SMART59791.2023.10428163.

[21] J. M. Martín-Doñas and A. Álvarez, "The vicomtech audio deepfake detection system based on wav2vec2 for the 2022 add challenge," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2022, pp. 7937–7941. doi: 10.1109/ICASSP43922.2022.9747768.

[22] A. Khan, K. M. Malik, and S. Nawaz, "Frame-to-utterance convergence: a spectra-temporal approach for unified spoofing detection," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2024, pp. 10761–10765. doi: 10.1109/ICASSP48485.2024.10447500.

[23] J. Frank and L. Schönherr, "WaveFake: A data set to facilitate audio deepfake detection," *Adv. Neural Inf. Process. Syst.*, no. NeurIPS, 2021.

[24] K. Thakral, H. Agarwal, K. Naraya, S. Mittal, M. Vatsa, and R. Singh, "DeePhyNet: towards detecting phylogeny in deepfakes," *IEEE Trans. Biometrics, Behav. Identity Sci.*, 2024, doi: 10.1109/TBIOM.2024.3487482.

[25] J. Yang, R. K. Das, and H. Li, "Significance of subband features for synthetic speech detection," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 2160–2170, 2020, doi: 10.1109/TIFS.2019.2956589.

[26] Y. Zhang, J. Lu, Z. Li, Z. Shang, W. Wang, and P. Zhang, "Improving the robustness of deepfake audio detection through confidence calibration," in *CEUR Workshop Proceedings*, 2023, pp. 70–75.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[27] S. A. Momu, R. R. Siddiqui, S. S. Shanto, and Z. Ahmed, "A comprehensive approach to deepfake audio detection: using feature fusion and deep learning," in *2024 27th International Conference on Computer and Information Technology,* ICCIT *2024 - Proceedings*, 2024, pp. 351–356. doi: 10.1109/ICCIT64611.2024.11022599.

[28] K. Schäfer and M. Steinebach, "MFCC vs. LFCC for audio deepfake detection: the role of delta features and input length," in *European Signal Processing Conference*, 2025, pp. 576–580. doi: 10.23919/EUSIPCO63237.2025.11226775.

[29] X. Wang *et al*., "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," 2020, Accessed: Jun. 25, 2026. [Online]. Available: https://www.asvspoof.org/index2019.html

[30] Z. K. Abdul and A. K. Al-Talabani, "Mel frequency cepstral coefficient and its applications: a review," *IEEE Access*, vol. 10, pp. 122136–122158, 2022, doi: 10.1109/ACCESS.2022.3223444.

[31] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with Rawnet2," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing* - Proceedings, 2021, pp. 6369–6373. doi: 10.1109/ICASSP39728.2021.9414234.

[32] K. Z. Mon, K. Galajit, C. O. Mawalim, J. Karnjana, T. Isshiki, and P. Aimmanee, "Spoof detection using voice contribution on LFCC features and ResNet-34," in *18th International Conference on Artificial Intelligence and Natural Language Processing and International Conference on Artificial Intelligence and Internet of Things,* iSAI-NLP 2023, 2023. doi: 10.1109/iSAI-NLP60301.2023.10354625.

[33] K. S. Krishnan and K. S. Krishnan, "MFAAN: Unveiling audio deepfakes with a multi-feature authenticity network," in *2023 9th International Conference on Signal Processing and Communication,* ICSC 2023, 2023, pp. 150–155. doi: 10.1109/ICSC60394.2023.10441405.

[34] R. Mahyavanshi, C. V Mahesh Reddy, A. J. Shah, and H. A. Patil, "Teager energy cepstral coefficients for audio deepfake detection," in *APSIPA ASC 2024* - Asia Pacific Signal and Information Processing Association Annual Summit and Conference 2024, 2024. doi: 10.1109/APSIPAASC63619.2025.10848893.

[35] J. Zhou and L. Yang, "Research on audio scene classification method based on deep learning technology in sound processing," in *ACM International Conference Proceeding Series*, 2024, pp. 664–668. doi: 10.1145/3675417.3675527.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[36] L. Z. Yong and H. Nugroho, "Acoustic anomaly detection of mechanical failure: time-distributed CNN-RNN deep learning models," in *Lecture Notes in Electrical Engineering*, 2022, pp. 662–672. doi: 10.1007/978-981-19-3923-5_57.

[37] L. Pham, P. Lam, T. Nguyen, H. Nguyen, and A. Schindler, "Deepfake audio detection using spectrogram-based feature and ensemble of deep learning models," in *IEEE 5th International Symposium on the Internet of Sounds, IS2 2024*, 2024. doi: 10.1109/IS262782.2024.10704095.

[38] K. Li, X.-M. Zeng, J.-T. Zhang, and Y. Song, "Convolutional recurrent neural network and multitask learning for manipulation region location," in *CEUR Workshop Proceedings*, 2023, pp. 18–22.

[39] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing attack detection using the non-linear fusion of sub-band classifiers," in *Proceedings of the Annual Conference of the International Speech Communication Association,* INTERSPEECH, 2020, pp. 1106–1110. doi: 10.21437/Interspeech.2020-1844.

[40] R. Lomte, P. Singhal, S. Patil, Z. Shaikh, and P. Agarwal, "A multi-scale residual network with hierarchical feature fusion for robust audio deepfake detection," in *Lecture Notes in Networks and Systems*, 2026, pp. 449–466. doi: 10.1007/978-3-032-13803-3_36.

[41] K. Galajit *et al.*, "ThaiSpoof: A database for spoof detection in Thai language," in *18th International Conference on Artificial Intelligence and Natural Language Processing and International Conference on Artificial Intelligence and Internet of Things,* iSAI-NLP 2023, 2023. doi: 10.1109/iSAI-NLP60301.2023.10354956.