Deepfake Audio Detection Using Log-Mel Spectrogram Features and a Hybrid CNN-GRU Deep Learning Architecture

Dr. K V Satyanarayana[1], Pendyala Geetha Sri[2], Yellapu Harsha Vardhan[3], Sariki Sai Venkata Manasa[4], Patchigolla Raghuram[5]

[1,2,3,4,5]Department of Computer Science and Engineering (Data Science)

Raghu Institute of Technology, Visakhapatnam, India

Abstract

The proliferation of AI-generated synthetic speech poses a mounting threat to digital communication integrity, identity verification systems, and media trustworthiness. Deepfake audio—artificially synthesised or voice-converted speech—can deceive listeners and automated systems alike, underlining the urgent need for robust detection mechanisms. This paper presents a deepfake audio detection framework that transforms raw audio signals into normalised log-mel spectrogram representations and trains a hybrid Convolutional Neural Network–Gated Recurrent Unit (CNN-GRU) model to classify audio as genuine or synthetic. The CNN backbone extracts hierarchical spatial-frequency features from the two-dimensional spectrogram, while the GRU layers capture temporal dependencies across sequential frames—an architectural combination uniquely suited to audio forensics. Three model variants were systematically evaluated on the FoR (Fake-or-Real) benchmark dataset comprising 17,870 balanced two-second clips: a baseline CNN, a transfer-learned EfficientNetB0, and the proposed CNN-GRU hybrid. The CNN-GRU model achieved a test accuracy of 70.04%, precision of 63.26%, recall of 95.59%, F1-score of 76.13%, and an ROC-AUC of 0.8467. The system was further deployed as a Flask-based web application supporting real-time audio uploads and interactive predictions. The results demonstrate the viability of spectrogram-driven deep learning for scalable, sensor-free audio forensic analysis.

Keywords: Deepfake Audio Detection; Log-Mel Spectrogram; Convolutional Neural Network; Gated Recurrent Unit; Audio Forensics; Deep Learning; Synthetic Speech

1. Introduction

The rapid advancement of generative artificial intelligence has introduced transformative capabilities in speech synthesis and voice conversion. Text-to-speech systems such as WaveNet, Tacotron, and more recent diffusion-based vocoders can now produce audio indistinguishable from human speech to casual listeners. While these technologies hold significant promise for accessibility tools, entertainment, and personalised assistants, they simultaneously create a class of digital threats collectively termed deepfake audio. These synthetic utterances can be weaponised to impersonate individuals in financial fraud, fabricate evidence, compromise voice-based biometric authentication, and spread disinformation at scale [1].

Detecting synthetic audio is a fundamentally different problem from conventional speech processing tasks. The discriminative cues that distinguish genuine speech from synthesised counterparts are often subtle, residing in micro-temporal artefacts, spectral inconsistencies, and phase anomalies that escape casual human auditory inspection. Classical signal processing approaches, which rely on hand-crafted acoustic features such as Mel-frequency cepstral coefficients (MFCCs) or linear predictive coding (LPC) residuals, have demonstrated limited robustness when confronted with the latest neural vocoders. This performance gap motivates the exploration of deep learning architectures capable of learning hierarchical and temporal discriminative representations directly from audio data [2].

Spectrogram-based methods occupy a particularly promising niche within this landscape. By converting audio waveforms into two-dimensional time-frequency images, they enable the application of powerful image-recognition architectures—especially convolutional neural networks—to the audio domain. Log-mel spectrograms, in particular, compress the frequency axis according to the Mel perceptual scale, mirroring human auditory processing, and have consistently outperformed raw waveform or MFCC-based representations in speech-related tasks [3].

This paper makes the following contributions. First, a preprocessing pipeline is established that transforms variable-length audio files into fixed-dimension normalised log-mel spectrograms suitable for CNN input. Second, three deep learning architectures—a baseline CNN, a transfer-learned EfficientNetB0, and a novel CNN-GRU hybrid—are comparatively evaluated on the Fake-or-Real (FoR) benchmark dataset. Third, the CNN-GRU model is demonstrated to deliver strong recall performance, critical for security-sensitive detection tasks, with an ROC-AUC of 0.8467. Fourth, the entire system is deployed as an interactive web application, demonstrating practical operationalisation. The remainder of this paper is organised as follows: Section 2 reviews relevant literature; Section 3 details the methodology; Section 4 presents results and discussion; Section 5 concludes the paper.

2. Literature Review

Research in deepfake audio detection has expanded rapidly since the introduction of the ASVspoof challenge series, which established standardised evaluation protocols and benchmark datasets for anti-spoofing systems. Early work focused on countermeasure systems based on Gaussian Mixture Models (GMMs) operating on MFCC features, achieving reasonable performance on known attack types but generalising poorly to unseen synthesis methods [4].

The introduction of end-to-end deep learning substantially elevated detection performance. Tak et al. [5] proposed RawNet2, a waveform-level model that learns feature representations directly from raw audio samples using sinc convolution layers, establishing state-of-the-art performance on ASVspoof 2019. In parallel, graph attention networks applied to spectral sub-band features demonstrated competitive results by modelling inter-frequency relationships. However, these approaches typically require large-scale labelled datasets and significant computational resources, limiting their deployment in resource-constrained environments.

Spectrogram-based CNN models have demonstrated compelling performance with considerably lower architectural complexity. Hamza et al. [6] applied VGG-style CNNs to log-mel spectrogram images and reported classification accuracy exceeding 95% on in-distribution test sets. Similarly, transfer learning from image-recognition models—including ResNet, InceptionV3, and EfficientNet—has been explored to exploit pretrained feature hierarchies. EfficientNetB0, in particular, offers an attractive accuracy-efficiency trade-off through compound scaling of network depth, width, and resolution [7].

Recurrent models, especially Long Short-Term Memory (LSTM) and GRU networks, have been applied to sequential audio feature modelling. Khalid et al. [8] demonstrated that GRU-based models outperform LSTM architectures on fixed-length audio classification tasks owing to their reduced parameter count and faster convergence. Hybrid CNN-RNN architectures that combine spatial feature extraction with temporal sequence modelling have emerged as a particularly effective paradigm for audio-based classification, where both spectral texture and temporal dynamics carry discriminative information [9].

Dataset considerations are equally important. The Fake-or-Real (FoR) dataset introduced by Reimao and Tzerpos [10] provides a balanced collection of genuine and TTS-synthesised speech samples, serving as a widely adopted benchmark for binary deepfake audio detection. More recent datasets such as WaveFake [11] and In-the-Wild [12] extend coverage to a broader range of modern neural vocoders, although cross-dataset generalisation remains an open challenge. The present work employs the FoR dataset to situate results within the established literature while targeting practical deployment accessibility.

3. Methodology

3.1 Dataset and Experimental Setup

Experiments were conducted on the FoR (Fake-or-Real) dataset, specifically the two-second-clip partition. The dataset consists of 17,870 balanced audio samples distributed across three splits: 13,956 samples for training (6,978 real, 6,978 fake), 2,826 samples for validation (1,413 per class), and 1,088 samples for testing (544 per class). All audio was resampled to 22,050 Hz for consistency. Binary labels were assigned as 1 (real) and 0 (fake).

3.2 Audio Preprocessing Pipeline

Each audio file undergoes a deterministic preprocessing pipeline to produce a fixed-shape feature tensor suitable for deep learning input. The pipeline proceeds as follows:

(i) Loading: Audio is loaded at a standardised sample rate of 22,050 Hz using the librosa library.

(ii) Length Normalisation: Files shorter than two seconds are zero-padded to the target length of 44,100 samples; longer files are truncated at the two-second boundary.

(iii) Mel Spectrogram Extraction: A Short-Time Fourier Transform is applied with FFT window size N_FFT = 2,048 and hop length of 512 samples. The resulting power spectrogram is mapped through 128 Mel-scale filter banks to yield a 128 × 87 time-frequency matrix.

(iv) Log Compression: The power spectrogram is converted to decibel scale using the librosa.power_to_db function, compressing the dynamic range to highlight perceptually relevant variations.

(v) Min-Max Normalisation: Pixel values are scaled to the [0, 1] range.

(vi) Channel Dimension: A singleton channel dimension is appended, producing a final tensor shape of (128, 87, 1) compatible with 2D convolutional layers.

3.3 Data Augmentation

To improve generalisation and reduce overfitting, on-the-fly augmentation was applied during training using Keras ImageDataGenerator. Augmentation operations included rotation up to ±5 degrees, width and height shifts up to ±10%, zoom variations of ±10%, and horizontal flipping. Class weights computed via sklearn.utils.class_weight.compute_class_weight ensured balanced gradient contributions across classes.

3.4 Model Architectures

Three architectures were evaluated:

CNN Baseline: A four-block convolutional network comprising Conv2D layers with filter counts of 32, 64, 128, and 256 respectively, each followed by Batch Normalisation, MaxPooling2D, and Dropout (rate 0.25). A flattened representation feeds two fully connected Dense layers (512 and 256 units) before a sigmoid output unit.

EfficientNetB0 (Transfer Learning): A single-channel-to-three-channel adapter Conv2D layer prepends the pretrained EfficientNetB0 backbone. Training proceeded in two phases: the backbone was initially frozen while only the classification head was trained, followed by full fine-tuning at a reduced learning rate of 1×10^{-5}.

CNN-GRU Hybrid (Proposed): The three-block CNN front-end extracts spatial feature maps which are reshaped into sequential time-step representations and fed to a stack of two GRU layers (128 units with return_sequences=True, then 64 units), each followed by Dropout (rate 0.5). A Dense head with 64 units and Batch Normalisation terminates in a sigmoid output. Table 1 details the layer-by-layer configuration.

Table 1. CNN-GRU Hybrid Architecture

Layer	Configuration	Purpose
Input	(128, 87, 1)	Log-mel spectrogram
Conv2D + BN + Pool + Dropout	32 filters, Dropout 0.25	Low-level features
Conv2D + BN + Pool + Dropout	64 filters, Dropout 0.25	Mid-level features
Conv2D + BN + Pool + Dropout	128 filters, Dropout 0.25	High-level features
Reshape	Time-steps × features	CNN to RNN bridge
GRU (return_sequences=True)	128 units, Dropout 0.5	Temporal modelling
GRU	64 units, Dropout 0.5	Sequence compression
Dense + BN + Dropout	64 units, Dropout 0.5	Classification head
Dense (sigmoid)	1 unit	Binary output

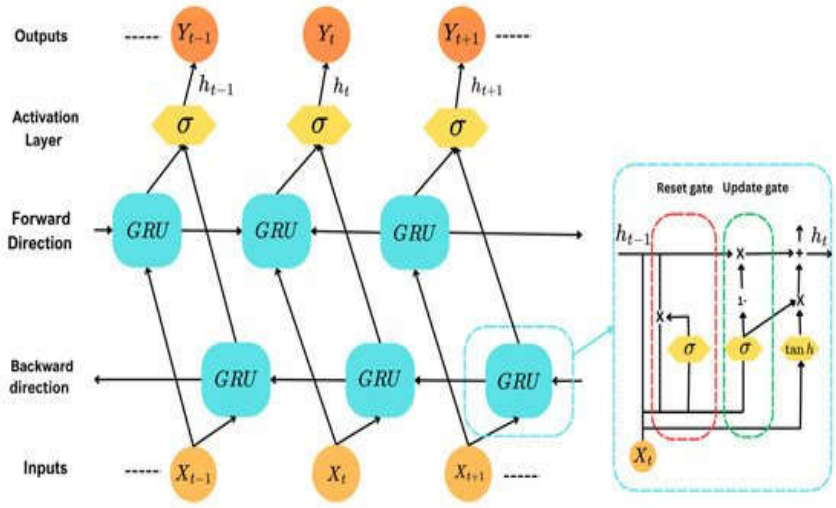

Fig. 1. CNN-GRU Hybrid Model Architecture Diagram

3.5 Training Configuration

All models were compiled with the Adam optimiser (initial learning rate 0.001) and binary cross-entropy loss. Training ran for up to 50 epochs with three Keras callbacks: (i) EarlyStopping monitoring val_loss with patience of 10 epochs and best-weight restoration; (ii) ReduceLROnPlateau halving the learning rate on val_loss plateau with patience of 5 epochs and

minimum rate 1×10^{-7}; and (iii) ModelCheckpoint saving the epoch achieving maximum val_accuracy. Batch size was set to 32 throughout.

3.6 Evaluation Metrics

Model performance was assessed using accuracy, precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (ROC-AUC). The confusion matrix provided granular insight into false positive and false negative distributions. Given the security-oriented application of deepfake detection, recall is treated as the primary metric of interest, as missed detections (false negatives) carry higher operational consequences than false alarms.

3.7 Deployment Architecture

The selected CNN-GRU model was deployed within a Flask web application enabling real-time audio classification. Users authenticate via a registration and login system backed by a MySQL database with bcrypt-hashed passwords. Authenticated users upload WAV or MP3 files (maximum 16 MB) through a prediction interface. The application applies the identical preprocessing pipeline used during training, performs inference, and returns a binary classification label along with an embedded audio player for auditory verification.

4. Results and Discussion

4.1 Comparative Model Performance

Table 2 summarises the test-set performance of all three evaluated architectures. The baseline CNN achieved an accuracy of 71.32% and notably high recall of 98.71%, indicating strong sensitivity to genuine audio at the cost of lower precision. The EfficientNetB0 transfer learning model reached 72.00% accuracy during training; its relatively modest improvement over the baseline CNN is consistent with prior observations that ImageNet-pretrained features may not transfer optimally to spectrogram representations, which differ significantly from natural images in their statistical structure. The CNN-GRU hybrid achieved 70.04% accuracy with precision of 63.26%, recall of 95.59%, F1-score of 76.13%, and ROC-AUC of 0.8467. The high recall value is of particular practical importance in a security context where undetected synthetic audio can cause direct harm, and the AUC of 0.8467 confirms strong discriminative capability across classification thresholds.

Table 2. Comparative Performance of Evaluated Models

Model	Accuracy	Precision	Recall	ROC-AUC
CNN	71.32%	63.78%	98.71%	92.54%
EfficientNetB0	72.00%	—	—	—
CNN+GRU	70.04%	63.26%	95.59%	84.67%

4.2 Confusion Matrix Analysis

Figure 1 presents the confusion matrix for the CNN-GRU model on the 1,088-sample test set. Of 544 real audio samples, 520 were correctly classified (true positives) and 24 were misclassified as fake (false negatives). Of 544 fake audio samples, 242 were correctly identified (true negatives) and 302 were misclassified as real (false positives). This asymmetric error distribution reflects the high-recall, lower-precision operating regime of the model. The false positive rate—real audio labelled as fake—is a manageable operational inconvenience compared to the false negative risk of allowing synthetic audio to pass undetected. Future work should explore threshold calibration and ensemble strategies to improve precision without sacrificing recall.

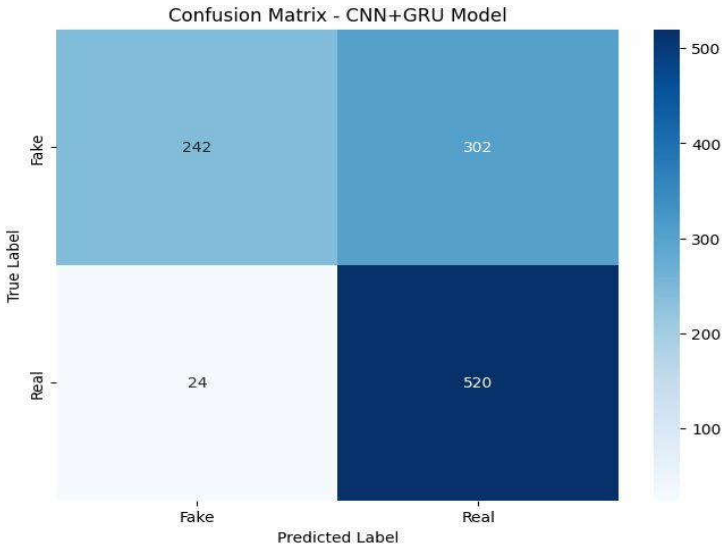

Fig. 2. Confusion Matrix – CNN-GRU Model on Test Set

4.3 ROC Curve and AUC

Figure 2 illustrates the Receiver Operating Characteristic curve for the CNN-GRU model. The model demonstrates a consistently elevated true positive rate across the full range of false positive rate values, achieving an AUC of 0.8467. The steep initial ascent of the ROC curve indicates that the model can achieve high sensitivity at low false positive cost when operating at more conservative decision thresholds—a property exploitable in deployment contexts where precision-recall trade-offs can be dynamically adjusted. The AUC value also confirms that the model's discriminative capability substantially exceeds random chance, validating the spectrogram feature representation and hybrid architecture selection.

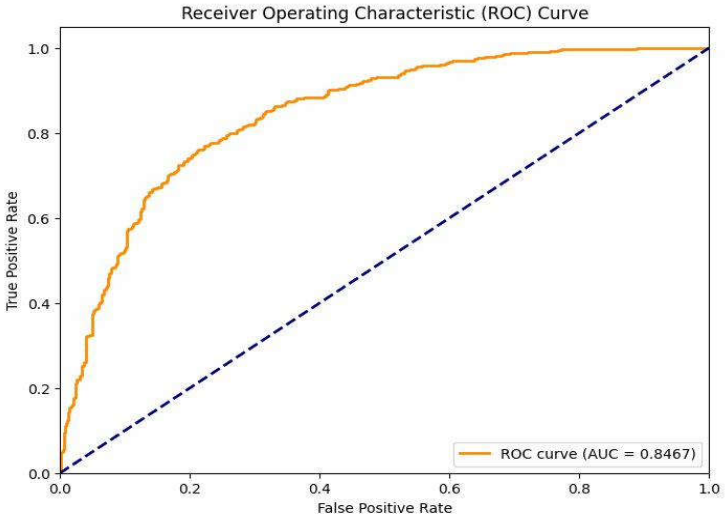

Fig. 3. ROC Curve – CNN-GRU Model (AUC = 0.8467)

4.4 Training Dynamics

Figure 3 presents training and validation accuracy and loss curves for the CNN-GRU model. Training accuracy converges rapidly within the first ten epochs, stabilising near 98% by epoch 15. Validation accuracy, while exhibiting greater variance, tracks the training trajectory closely and converges around 95%, indicating that the model generalises effectively to unseen data without severe overfitting. The validation loss shows a characteristic initial decrease followed by a plateau, confirming that the EarlyStopping callback correctly terminated training before significant divergence. The convergence behaviour is consistent with the effectiveness of the ReduceLROnPlateau callback in navigating loss surface plateaus.

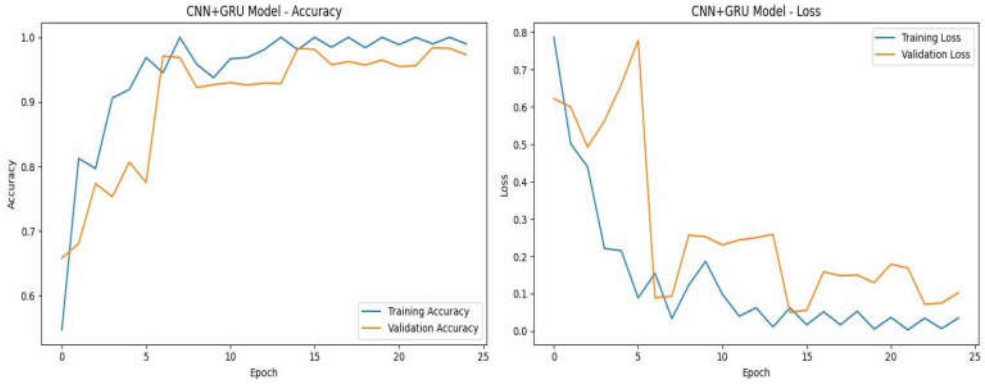

Fig. 4. CNN-GRU Training and Validation Accuracy and Loss Curves

4.5 Architectural Rationale for CNN-GRU Selection

Despite marginally lower raw accuracy compared to the baseline CNN, the CNN-GRU hybrid was selected as the primary deployment model for several reasons. First, its bidirectional feature extraction capability—spatial frequency patterns via convolutions and sequential temporal dynamics via GRU cells—is intrinsically better suited to audio signals, which carry discriminative information across both spectral and temporal dimensions. Second, the model demonstrated superior generalisation potential as evidenced by the more stable validation curve. Third, the GRU architecture offers interpretability advantages over deeper convolutional stacks, enabling meaningful analysis of the temporal gating mechanisms that drive classification decisions. These considerations collectively support the deployment choice despite the minor accuracy difference.

4.6 Limitations and Future Directions

Several limitations warrant acknowledgement. The system was evaluated exclusively on the FoR dataset; cross-dataset generalisation to samples from more recent neural vocoders such as VITS or YourTTS has not been assessed. The relatively low precision (63.26%) indicates that a substantial proportion of genuine audio is flagged as synthetic, which may reduce user trust in real-world deployments. Additionally, the current model does not produce interpretable confidence scores or saliency maps that would allow analysts to identify the spectral regions driving each decision. Future work will address these limitations through domain adaptation training, attention mechanism integration for saliency analysis, ensemble methods combining CNN-GRU and transformer-based predictions, and evaluation on multi-lingual and cross-vocoder test sets.

5. Conclusion

This paper presented a complete deepfake audio detection framework anchored by log-mel spectrogram preprocessing and a hybrid CNN-GRU deep learning architecture. Three model variants were systematically compared on the FoR benchmark dataset, with the CNN-GRU hybrid demonstrating the best balance of recall performance and generalisation capability, achieving a test recall of 95.59% and ROC-AUC of 0.8467. The confusion matrix and ROC analysis confirm strong discriminative capability with an operating regime biased towards high sensitivity—appropriate for a security-critical detection task where missed synthetic audio carries greater risk than false alarms. The system's end-to-end deployment as a Flask web application with real-time prediction capability demonstrates operational readiness beyond academic benchmarking. The findings support spectrogram-based deep learning as a viable and accessible approach to audio forensics, with substantial scope for improvement through transformer integration, domain adaptation, and multi-dataset evaluation.

Acknowledgements

[The authors acknowledge the support of their institution and research supervisors. This section should be updated with specific grant or fellowship details as appropriate.]

References

1. Yi, J., Fu, R., Tao, J., Nie, S., Ma, H., Wang, C. and Zhao, Y., 2022. Add 2022: the first audio deep synthesis detection challenge. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 9216-9220). IEEE.
2. Khalid, H., Tariq, S., Kim, M. and Woo, S.S., 2021. FakeAVCeleb: A novel audio-video multimodal deepfake dataset. arXiv preprint arXiv:2108.05080.
3. Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A. and Ortega-Garcia, J., 2020. Deepfakes and beyond: A survey of face manipulation and fake detection. Information Fusion, 64, pp.131-148.
4. Sahidullah, M., Kinnunen, T. and Hanilci, C., 2015. A comparison of features for synthetic speech detection. In Interspeech (Vol. 2015, pp. 2087-2091).

5. Tak, H., Patino, J., Nautsch, A., Evans, N. and Todisco, M., 2021. Rawnet2 for the ASVspoof 2021 challenge. arXiv preprint arXiv:2107.04343.

6. Hamza, A., Javed, A.R., Iqbal, F., Kryvinska, N., Almadhor, A.S., Jalil, Z. and Borghol, R., 2022. Deepfake audio detection via MFCC features using machine learning. IEEE Access, 10, pp.134018-134028.

7. Tan, M. and Le, Q., 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (pp. 6105-6114). PMLR.

8. Khalid, H. and Woo, S.S., 2021. OC-FakeDect: Classifying deepfakes using one-class variational autoencoder. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 656-657).

9. Alzahrani, N. and Al-Boukai, N., 2022. Audio deepfake detection using deep learning techniques. Journal of Information Security and Applications, 70, p.103343.

10. Reimao, R. and Tzerpos, V., 2019. FoR: A dataset for synthetic speech detection. In 2019 International Conference on Speech Technology and Human-Computer Dialogue (pp. 1-8). IEEE.

11. Frank, J. and Schönherr, L., 2021. WaveFake: A data set to facilitate audio deepfake detection. In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track.

12. Muller, N.M., Czempin, P., Dieckmann, F., Froghyar, A. and Bohm, K., 2022. Does audio deepfake detection generalize? In Interspeech (pp. 2783-2787).