Biometric Signatures: Leveraging Audio and Behavioral Traits for Deepfake Detection

Karmishtha Patnaik
SOCSE (BTech)
RV University
Bengaluru, Karnataka
karmishthap.btech22@rvu.edu.in

Renu Bojja
SOCSE (BTech)
RV University
Bengaluru, Karnataka
renub.btech22@rvu.edu.in

Manjul Krishna Gupta
SOCSE (BTech)
RV University
Bengaluru, Karnataka
manjulkrishnag@rvu.edu.in

Abstract—Verifying the authenticity of digital content is increasingly critical as deepfake technology leverages deep learning to generate highly convincing synthetic media. In this study, we introduce a biometric-driven detection framework that combines audio features with behavioral-anatomical markers to boost deepfake detection accuracy. This framework analyzes unique voice characteristics—such as pitch, timbre, and spectral properties—alongside individual-specific behaviors, capturing subtle, hard-to-replicate biometric signals for enhanced detection robustness. This dual-focus approach, integrating voice and behavior-based anatomical analysis, provides a resilient solution against digital deception in practical applications.

Our detection framework utilizes the Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM), a model specifically designed for deepfake clips because it is capable of detecting spatial and temporal anomalies. The CNN layers are designed to learn spatial features within each video frame and detect frame-level discrepancies, which are indicative of deepfakes. These spatial features are then passed through LSTM layers where the changes in spatial features over different frames are then identified. This is especially helpful for detecting anomalies such as blinkers, motion that seems erratic from one frame to the next or inconsistent facial expressions that cannot realistically be reproduced across frames successively. The configuration of the CNN-LSTM model helps to improvise the detection performance by providing the spatial resolution with the temporal resolution, which gives an idea about the depth of the anomalies in deepfake videos. Moreover, the ability of the model to work with sequential data provides a high level of resistance to adversarial procedures that are designed to overcome classical CNN-only models through frame-specific manipulations. Through considerations of feature visuals as well as temporal sequencing, CNN-LSTM greatly enhances the detection credibility and offers a flexible and expandable solution for applying deepfake detection in various fields.

This study explores the use of biometric audio signatures for detecting deepfake audio. With the rise of sophisticated voice synthesis technologies, detecting audio deepfakes has become a critical concern in cybersecurity, media, and communication sectors. By leveraging both traditional and deep learning-based models, specifically K-Nearest Neighbors (KNN) and Bidirectional Long Short-Term Memory (BiLSTM) networks, this research investigates the unique characteristics of biometric audio features that can distinguish real from synthetic voices. Spectral features such as Mel-frequency Cepstral Coefficients (MFCCs), chroma, and Tonnetz are extracted and analyzed, with Generative Adversarial Networks (GANs) further augmenting the dataset to enhance model robustness. A comparative spectrogram analysis highlights unique spectral patterns in real versus fake audio samples, underscoring the critical role of biometric features in audio deepfake detection. Findings reveal that BiLSTM, augmented with GAN-synthesized data, outperforms traditional methods in accuracy and robustness, with significant implications for future audio-based security systems.

Keywords— Deepfake, biometric-driven detection, biometric audio signatures, synthetic media, CNN-LSTM, GAN, K-Nearest Neighbors, BiLSTM, MFCC, chroma, Tonnetz, spectrogram analysis, media authenticity, digital security.

I. Introduction

The development of artificial intelligence (AI) and machine learning (ML) has led to rapid advancements in audio and visual synthesis technologies, including the creation of “deepfakes.” Deepfakes utilize AI algorithms to fabricate or modify audio and visual data to convincingly mimic real-world individuals. These technologies, initially developed for creative and entertainment purposes, are now increasingly used in malicious applications, such as identity theft, fraud, and misinformation campaigns. In particular, audio deepfakes pose unique challenges due to the fidelity with which they can replicate an individual's voice. By manipulating subtle vocal nuances, deepfake audio can deceive even advanced authentication systems.

Traditional deepfake detection research has predominantly focused on image and video manipulation. However, with the increasing accessibility of advanced text-to-speech (TTS) technologies and voice cloning software, audio deepfake detection has emerged as a vital area of research. Synthetic audio generated by advanced TTS systems like WaveNet and Tacotron 2 can closely mimic real human voices, often bypassing existing security measures. These systems utilize neural networks to analyze and replicate the vocal characteristics of an individual, producing speech that is almost indistinguishable from authentic recordings. Consequently, detecting such synthetic audio requires sophisticated models that can recognize and differentiate subtle biometric patterns inherent in genuine human voices.

In response to these challenges, this research paper examines biometric audio signatures as a reliable solution for detecting audio deepfakes. Biometric audio features, unique to each individual, encompass vocal characteristics like pitch, rhythm, and frequency distribution. By focusing on these characteristics, the study aims to identify patterns that are difficult for TTS systems to replicate, providing a robust method to distinguish real from synthetic audio. Two machine learning models, K-Nearest Neighbors (KNN) and Bidirectional Long Short-Term Memory (BiLSTM) networks, are employed to evaluate the effectiveness of these biometric signatures in detecting deepfake audio. KNN serves as a baseline for understanding how traditional models handle static biometric features, while BiLSTM is utilized for its ability to process sequential data, capturing the temporal dependencies in audio that are crucial for deepfake detection.

Furthermore, the study incorporates Generative Adversarial Networks (GANs) to augment the dataset by generating diverse synthetic audio samples that improve the robustness of the BiLSTM model. GANs play a dual role by both aiding in data generation for model training and serving as a core technique in deepfake video creation. In video-based deepfake detection, hybrid architectures like Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) models are essential. CNN layers capture spatial features within frames—detecting misalignments or irregular textures—while LSTM layers analyze temporal dependencies across frames to reveal abrupt transitions or unnatural motion. This approach strengthens detection by integrating spatial and temporal analysis, making CNN-LSTM models highly effective for real-time verification in applications like social media monitoring, forensic analysis, and video-based security.

Spectral analysis, specifically through Mel-frequency Cepstral Coefficients (MFCCs), chroma, and Tonnetz features, forms the core of this biometric-based approach. By comparing spectrograms of real and synthetic audio, the research identifies distinct spectral patterns that characterize deepfake audio, enhancing detection accuracy and offering a foundation for future research in biometric-based audio verification systems.

II. LITERATURE SURVEY

A. Deep Learning in Audio Classification

Deep learning has significantly transformed the field of audio processing, particularly in applications like speech recognition, speaker identification, and emotion detection. Recurrent neural networks (RNNs), especially Long Short-Term Memory (LSTM) networks, have proven effective in modeling sequential data, which is essential for audio analysis. LSTM networks can capture long-term dependencies, allowing them to process sequential inputs, such as audio waveforms, where the temporal context is critical. Bidirectional LSTM (BiLSTM) models, an advanced variant of LSTM, allow information to be processed in both forward and backward directions. This capability is particularly useful in deepfake detection, where the model needs to identify complex vocal patterns that may not be apparent from a single direction of analysis.

Previous studies have demonstrated the efficacy of LSTM and BiLSTM in capturing sequential audio features for various tasks. For instance, in speaker identification, these models can distinguish between different speakers by analyzing temporal changes in vocal characteristics. In the context of deepfake detection, BiLSTM's ability to retain sequential information provides an advantage over traditional models, as it can recognize subtle irregularities in synthetic audio. Studies have highlighted the potential of BiLSTM in identifying synthetic patterns within audio, particularly when used in conjunction with carefully selected biometric features.

B. Biometric Audio Features and Traditional Classification

Biometric audio signatures have been widely used in speaker verification and identification systems. These signatures encompass unique vocal characteristics, including pitch, tone, and frequency distribution, which vary from person to person. Mel-frequency Cepstral Coefficients (MFCCs) are one of the most commonly used features in biometric audio applications due to their effectiveness in representing the spectral properties of the human voice. MFCCs capture the power spectrum of sound and provide a compact representation of an individual’s vocal characteristics. Additionally, chroma and spectral contrast features are often used in audio processing to capture harmonic content and amplitude differences, respectively. These features can highlight unique vocal nuances that may be difficult for synthetic models to replicate accurately.

The K-Nearest Neighbors (KNN) algorithm, a traditional machine learning model, has been employed extensively in speaker identification tasks due to its simplicity and interpretability. KNN operates by measuring the similarity between feature vectors, classifying samples based on the majority class of their nearest neighbors. Although effective in static pattern recognition, KNN’s reliance on fixed distance metrics limits its ability to capture the temporal variations inherent in audio sequences. In biometric audio-based deepfake detection, KNN can serve as a baseline, offering insights into the fundamental differences between real and synthetic audio patterns. However, it lacks the sequential processing capabilities required to detect the temporal dependencies in vocal features that advanced models like BiLSTM can capture.

C. Video Deepfake Detection Techniques

The emergence of video deepfake technology has prompted extensive research into detection methods due to the increasing sophistication of synthetic manipulations. The proliferation of realistic deepfake content raises significant concerns regarding misinformation, privacy, and content authenticity. Recent studies have underscored the effectiveness of hybrid architectures, particularly Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) models, in enhancing the accuracy of deepfake detection by integrating both spatial and temporal analyses.

Convolutional Neural Networks (CNNs) are proficient in capturing spatial inconsistencies within individual video frames. They excel at identifying artifacts characteristic of deepfake media, such as unnatural textures, inconsistent lighting, and misaligned facial features. These spatial discrepancies serve as key indicators in the classification process, allowing models to effectively discern real videos from manipulated ones. Research has demonstrated that CNNs can efficiently extract features that highlight anomalies indicative of synthetic alterations, providing a robust foundation for further analysis.

Complementing the spatial capabilities of CNNs, Long Short-Term Memory (LSTM) networks focus on the temporal aspects of video data by modeling dependencies across sequential frames. LSTMs are particularly adept at recognizing anomalies related to motion and facial dynamics, such as erratic eye blinks, abrupt changes in expressions, and unnatural movement patterns. These temporal anomalies are critical for identifying inconsistencies that may not be apparent within individual frames. The dual-layered approach of the CNN-LSTM architecture allows for a comprehensive analysis, addressing the shortcomings of single-method detection systems.

Several studies have demonstrated the efficacy of this hybrid approach in various applications, including social media monitoring, forensic video analysis, and secure video communications. The integration of CNNs and LSTMs not only enhances the detection of subtle artifacts and temporal incoherence but also improves the overall reliability of authenticity verification in diverse contexts.

D. GANs for Data Augmentation

Generative Adversarial Networks (GANs) have become popular for data augmentation in deep learning, especially in domains with limited labeled data or high class imbalance. GANs consist of two neural networks—a generator and a discriminator—that work together to produce synthetic data that closely resembles real samples. The generator creates fake data samples, while the discriminator attempts to differentiate between real and fake samples. Through this adversarial process, GANs can generate synthetic data that enhances the training dataset, improving the model's robustness and generalizability.

In the context of audio deepfake detection, GANs can augment the dataset by generating synthetic audio samples with varied characteristics. This augmented dataset exposes the detection model to a wider range of synthetic audio patterns, making it more resilient to different types of deepfakes. GAN-generated audio data provides a valuable training resource for the BiLSTM model, allowing it to learn from synthetic examples that mimic real human vocal characteristics. Studies like demonstrate that GAN-augmented data can improve model performance, particularly in applications like speech recognition and emotion detection, where training data is often limited or imbalanced.

Furthermore, Generative Adversarial Networks (GANs) are extensively used for augmenting datasets, providing diverse synthetic samples that enhance model robustness and improve generalizability across various deepfake types. GAN-augmented datasets expose CNN-LSTM models to a broader spectrum of synthetic manipulation patterns, leading to higher accuracy in detection across real-world applications such as social media monitoring, forensic video verification, and secure video conferencing.

III. METHODOLOGY

A. Libraries and Tools

1. Librosa: Librosa is a Python library extensively used for audio analysis and feature extraction. It offers a range of functionalities for loading audio files, performing spectral transformations, and extracting critical features such as MFCCs, chroma, and Tonnetz. Librosa's librosa.feature.mfcc() function allows for the extraction of MFCCs, which represent the power spectrum of sound, while librosa.feature.chroma_stft() and librosa.feature.tonnetz() enable the extraction of chroma and tonal centroid features, respectively. These features are vital for capturing the unique spectral and harmonic properties of audio that distinguish real from synthetic voices.
2. Pydub: Pydub simplifies audio manipulation, allowing for tasks such as format conversion, trimming, and segmentation. This study uses Pydub to preprocess audio samples, converting them into consistent 10-second segments. This segmentation ensures that all audio inputs are standardized, facilitating efficient processing and feature extraction.
3. NumPy and Pandas: NumPy provides essential numerical functions, particularly for handling large arrays of audio data, while Pandas supports data manipulation and analysis. These libraries play a crucial role in organizing audio features and preparing the data for model training and evaluation.
4. Scikit-learn: Scikit-learn is used for implementing the KNN classifier and performing hyperparameter optimization. Scikit-learn's GridSearchCV function is employed to optimize KNN's parameters, including the number of neighbors (n_neighbors), weighting (weights), and distance metric (p). Scikit-learn also provides evaluation metrics, such as accuracy, precision, recall, and F1-score, which are used to assess model performance.
5. OpenCV: OpenCV is an open-source computer vision library. Here, cv2 is used for reading, processing, and manipulating video frames, such as resizing and normalizing images. In deepfake detection, it helps prepare video frames for CNN processing.
6. Tensorflow: TensorFlow is a powerful deep learning framework. It is used here for building and training the CNN-LSTM model that detects deepfakes by identifying spatial and temporal inconsistencies in video data.

7. Seaborn and Matplotlib: Visualization libraries like Seaborn and Matplotlib are essential for data exploration and results presentation. They enable the creation of spectrograms and confusion matrices, providing insights into the classification performance of each model. Spectrogram visualizations, in particular, allow for the comparison of real and synthetic audio, highlighting spectral patterns that inform feature selection.
8. PyTorch: PyTorch serves as the primary deep learning framework for constructing and training the BiLSTM and GAN models. PyTorch's torch.nn module enables the creation of the BiLSTM architecture, while torch.optim provides optimizers like Adam for efficient training. PyTorch's flexibility and extensive support for GPU computation make it ideal for complex audio processing tasks.

B. Dataset Description

a. Real Audio Dataset

Real audio samples are sourced from public datasets like VoxCeleb and LibriSpeech, which provide authentic recordings across diverse speakers and speech contexts. These datasets contain thousands of hours of speech data, covering different languages, accents, and speaking styles. Each audio file is segmented into 10-second intervals, ensuring consistent input length and improving the reliability of feature extraction.

b. Synthetic Audio Dataset

Synthetic audio samples are generated using state-of-the-art TTS models, including Google's WaveNet and Tacotron 2, which are capable of producing highly realistic synthetic speech. These models use neural networks to analyze and replicate vocal characteristics, creating audio that closely mimics real human voices. By including synthetic audio samples from these TTS models, the dataset provides a range of deepfake audio that challenges the detection model, highlighting the importance of sophisticated feature extraction and classification.

c. Real Video Dataset

The FaceForensics++ dataset provides high-quality, real video samples specifically designed for forensic and deepfake research. This dataset includes diverse, high-resolution recordings with consistent lighting and various expressions, making it suitable for training and evaluating deepfake generation and detection models.

d. Synthetic Video Dataset

Deepfake videos are generated using advanced machine learning techniques, primarily through Generative Adversarial Networks (GANs) and autoencoders. Key methods include face-swapping, lip-syncing (e.g., LipGAN), and pose estimation (e.g., OpenPose) to create realistic facial expressions and movements. These techniques enable high-fidelity video synthesis by manipulating visual and audio inputs, creating convincingly altered or entirely fabricated videos.

C. Data Augmentation with GANs

GANs are employed to create additional synthetic samples, augmenting the dataset and exposing the model to a broader spectrum of fake audio patterns. The GAN-generated data introduces variations in pitch, timing, and noise, which improve the model's robustness against diverse synthetic audio. By training on both GAN-generated and real synthetic samples, the BiLSTM model becomes more resilient to the evolving sophistication of TTS models, allowing it to detect nuanced deepfake patterns.

GAN is utilized to synthetically generate diverse deepfake videos, expanding the dataset. By employing techniques such as face-swapping, lip-syncing, and pose estimation, GANs produce high-fidelity content that aids in identifying subtle manipulation anomalies. The CNN-LSTM architecture combines convolutional neural networks for spatial feature extraction with long short-term memory networks for temporal analysis, enhancing the model's detection capabilities through the enriched dataset.

IV. FEATURE EXTRACTION

Feature extraction is a critical step in the deepfake detection process, as it enables the identification of both spatial and temporal inconsistencies characteristic of manipulated media.

1. Spatial Feature Extraction using CNNs

Convolutional Neural Networks (CNNs) are used for spatial feature extraction in deepfake detection, focusing on frame-level anomalies like inconsistent lighting and facial irregularities. CNN layers process each video frame through convolution and pooling, capturing low-level details (e.g., edges) and complex textures. Convolutional operations generate feature maps that highlight synthetic artifacts, such as blurred edges or misaligned facial features. Max-pooling layers downsample these maps, optimizing computational efficiency while preserving critical spatial information. This approach enables accurate identification of deepfake artifacts within individual frames, enhancing the model's classification accuracy by leveraging robust spatial patterns across video sequences.

2. Temporal Feature Extraction using LSTMs

Long Short-Term Memory (LSTM) networks are adept at detecting temporal inconsistencies in deepfake videos, such as unnatural blinking or erratic head movements. Using memory cells, LSTMs retain sequential information, enabling frame-by-frame analysis of CNN-extracted features to detect continuity errors. This approach helps identify subtle artifacts, like inconsistent gaze or lighting shifts, indicative of synthetic manipulation. By analyzing each frame relative to prior ones, LSTMs effectively reveal deepfake anomalies. Combined with CNNs for spatial features, LSTM-based models enhance detection robustness against adversarial techniques, improving reliability in applications like forensic analysis and deepfake content verification.

3. Mel-frequency Cepstral Coefficients (MFCCs)

MFCCs are a core component in representing the spectral properties of audio. They capture information about the timbre of a voice, providing a compact representation that preserves the unique biometric traits of a speaker. In this study, MFCCs are computed using 13 coefficients per segment, with additional delta and delta-delta features capturing temporal changes. MFCCs have been widely used in speech recognition and are effective in distinguishing real from synthetic audio due to their sensitivity to spectral distortions.

4. Chroma and Spectral Contrast

Chroma features represent the harmonic content of audio, showing pitch-related information that is particularly relevant in speech analysis. Spectral contrast captures amplitude differences between peaks and valleys in the audio spectrum, highlighting distinct tonal characteristics. These features are crucial for identifying synthetic audio, as deepfake voices often exhibit uniform frequency distributions that lack the variability seen in authentic speech.

5. Tonal Centroid Features (Tonnetz)

Tonnetz features map harmonic properties of audio, reflecting changes in the tonality that are associated with speech dynamics. Real voices tend to show complex, varied tonal centroid patterns, while synthetic voices often lack these subtleties. By capturing these harmonic variations, Tonnetz features play a significant role in distinguishing real from fake audio, as they reveal inconsistencies in synthetic voices that are difficult to replicate accurately.

V. MODEL ARCHITECTURE AND IMPLEMENTATION

A. Hybrid Deep Learning : Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM)

The CNN-LSTM model combines the strengths of CNNs and LSTMs, providing a robust framework for deepfake detection by addressing both spatial and temporal dimensions in video data.

1. Architecture

The CNN-LSTM architecture utilizes Conv2D and MaxPooling2D layers to extract spatial features from individual frames, with TimeDistributed wrappers applying operations across each video frame. Conv2D filters detect pixel-level anomalies, while MaxPooling reduces dimensionality, enhancing computational efficiency. Flattening layers reformat spatial features for sequence processing by LSTM layers, which capture temporal inconsistencies in movement and expression. Dense layers follow for binary classification, with a sigmoid function in the output layer generating a probability score. A threshold of 0.5 classifies videos as real or fake, identifying deepfake content through both spatial and temporal cues.

2. Data Preprocessing and Generator Functions

Deepfake detection models require thorough preprocessing to manage the complexity of video data. This implementation includes resizing each frame to a fixed resolution and normalizing pixel values to [0, 1] to reduce computational load and enhance model convergence. Data generators supply labeled batches, with functions for real and fake data that extract frames and label sequences accordingly. Batches are shuffled to mix real and fake sequences, preventing class overfitting. These preprocessing steps standardize inputs, ensuring consistency and reliability during training and evaluation for effective deepfake detection.

3. Prediction Function

The prediction function leverages the CNN-LSTM model to analyze sequential frames from input videos, outputting a probability score for real-time deepfake detection. Convolutional Neural Network (CNN) layers extract spatial anomalies within each frame, while Long Short-Term Memory (LSTM) units capture temporal inconsistencies across sequences, identifying subtle manipulation patterns. This binary classification output enables efficient and accurate deepfake detection across applications, enhancing content authenticity in real-world contexts. The model's robust spatial-temporal analysis delivers high precision, supporting deployment in sensitive areas like social media monitoring, forensic analysis, and video conferencing security.

4. Training and Evaluation

The model is trained with binary cross-entropy loss, suitable for binary classification tasks like deepfake detection. Training occurs over multiple epochs, with steps_per_epoch controlling iterations per epoch. Hyperparameters such as learning rate, batch size, and epoch count are fine-tuned to maximize performance. During evaluation, accuracy and loss metrics on validation data help assess model generalization. High validation accuracy and low loss indicate strong generalizability, suggesting the model's effectiveness in detecting deepfakes in real-world scenarios. These metrics ensure the model is robust, capable of distinguishing real and manipulated content.

B. Baseline Model: K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) classifier functions as a baseline model, chosen for its simplicity and interpretability in measuring the effectiveness of biometric audio features in distinguishing real from synthetic audio. Unlike neural networks, KNN does not require extensive training, as it classifies audio samples based on similarity to labeled training samples. This characteristic makes KNN a suitable comparison point for understanding the limitations of traditional, non-sequential models in audio deepfake detection.

1. Feature Input and Data Preparation

   Audio features such as MFCCs, chroma, spectral contrast, and Tonnetz are used to form multi-dimensional feature vectors for each sample. These features capture both spectral and harmonic properties of audio, which are essential in differentiating between real and synthetic voices. In KNN, these features are stored in a feature matrix, where each row represents an audio sample and each column corresponds to a specific feature.

2. Distance Metric and Hyperparameter Tuning

   The KNN model relies on a distance metric, specifically Euclidean distance, to classify samples based on proximity to other samples in the feature space. Euclidean distance is effective for calculating straight-line distance between points in high-dimensional space, making it appropriate for MFCC-based audio features. To optimize the model, hyperparameters such as n_neighbors, weights, and algorithm are tuned through Scikit-learn's GridSearchCV:

a. n_neighbors: The optimal number of neighbors is selected based on cross-validation results, balancing the trade-off between model complexity and generalization.

b. weights: Both uniform and distance-based weighting schemes are evaluated. Uniform weights treat all neighbors equally, while distance weights assign more importance to closer neighbors, enhancing classification accuracy in clustered feature spaces.

c. algorithm: The choice between auto, ball_tree, kd_tree, and brute algorithms allows for computational efficiency depending on dataset size and feature dimensionality.

3. Training and Evaluation

   KNN, an instance-based learning algorithm, doesn't require traditional training. Instead, it classifies new audio samples by identifying the majority class among the k nearest neighbors. While KNN effectively leverages MFCCs and related features for basic audio classification, its static, distance-based approach fails to capture sequential dependencies, limiting its suitability for complex biometric applications that rely on understanding temporal patterns in audio data.

C. Deep Learning Model: Bidirectional Long Short-Term Memory (BiLSTM)

The Bidirectional Long Short-Term Memory (BiLSTM) network serves as the primary deep learning model in this study, capable of capturing sequential dependencies essential for accurately differentiating between real and synthetic audio. Unlike KNN, which processes data in static form, BiLSTM is designed to model temporal dynamics in audio signals, allowing it to learn from the sequential patterns inherent in human speech.

1. Architecture

   The BiLSTM model consists of two layers of bidirectional LSTM units, followed by dense layers for binary classification:

   a. Bidirectional LSTM Layers: The model uses two BiLSTM layers with 128 and 64 units, respectively. By processing input in both forward and backward directions, BiLSTM captures comprehensive contextual information, essential for detecting subtle vocal anomalies characteristic of synthetic audio.

   b. Dropout and Dense Layers: Dropout is applied to each BiLSTM layer to mitigate overfitting, with a rate of 0.3. Following the BiLSTM layers, fully connected (dense) layers with sigmoid activation provide binary classification outputs, indicating whether an audio sample is real or synthetic.

2. Feature Preparation for BiLSTM

   Unlike KNN, which treats feature vectors statically, BiLSTM processes audio features as sequences. MFCCs, chroma, and Tonnetz features are extracted from each segment, forming a time-series input where each time step represents a specific point in the audio's duration. These features are normalized and padded to a fixed length, ensuring uniformity across input sequences. The sequential nature of BiLSTM allows it to capture temporal fluctuations and subtle transitions in voice patterns that are often lacking in synthetic audio.

3. Training Procedure

   The BiLSTM model is trained on labeled real and synthetic audio samples, using binary cross-entropy as the loss function. Training is conducted over 50 epochs, with early stopping implemented to prevent overfitting. An 80/20 train-test split is used, ensuring that the model generalizes well to unseen samples. PyTorch's Adam optimizer is used for efficient gradient descent, with a learning rate of 0.001, providing a balance between convergence speed and stability.

D. Data Augmentation Using Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) are used to augment the dataset with additional synthetic audio samples. GANs consist of two neural networks—a generator and a discriminator—engaged in a competitive training process. In this context, the generator produces synthetic audio samples, while the discriminator learns to differentiate between real and fake samples. The GAN-generated samples provide variations in pitch, timing, and noise, which expose the BiLSTM model to a broader spectrum of synthetic patterns. This augmented dataset enables the BiLSTM model to improve its generalization capabilities, making it resilient to different types of audio manipulations.

VI. FEATURE EXTRACTION AND SPECTROGRAM ANALYSIS

A. Mel-frequency Cepstral Coefficients (MFCCs)

Mel-frequency Cepstral Coefficients (MFCCs) are widely recognized in audio analysis as a powerful feature representation of speech. By capturing the spectral characteristics of audio, MFCCs offer insights into the underlying patterns associated with the shape and movements of the vocal tract during speech production. These coefficients allow us to capture unique qualities of an individual's voice and the nuances of natural speech patterns, making MFCCs particularly valuable for distinguishing real audio from synthetic or generated audio.

1. Overview of MFCCs and Their Relevance

MFCCs are derived from the power spectrum of an audio signal, reflecting how energy is distributed across frequency bands that correspond to the perceptual Mel scale. The Mel scale is non-linear and closely aligned with human auditory perception, making it especially effective for capturing characteristics of speech. By compressing this spectral information into a few coefficients, MFCCs provide a compact representation that encapsulates the essence of the audio signal without requiring the full spectral detail. These coefficients have proven essential in applications like speech recognition, speaker identification, and audio classification.

2. Process of Extracting MFCCs

Extracting MFCCs begins by framing the audio signal and applying the Fast Fourier Transform (FFT) to shift it from time to frequency domain. These frequencies are mapped onto the Mel scale, emphasizing perceptually relevant ranges. A Mel spectrogram is created, with each frequency bin undergoing a logarithmic transformation to reflect human sensitivity to sound intensity. Finally, the Discrete Cosine Transform (DCT) is applied, producing MFCCs—a compact set of coefficients that capture essential spectral features like pitch, timbre, and formant structure.

3. Temporal Dynamics with Delta and Delta-Delta Coefficients

MFCCs capture the spectral details of each audio frame, but to account for speech's dynamic nature, delta (Δ) and delta-delta (ΔΔ) coefficients are also calculated. Delta coefficients track changes in MFCCs between frames, providing insight into spectral transitions, while delta-delta coefficients measure the acceleration of these changes, adding a second layer of temporal dynamics. Together, MFCCs, delta, and delta-delta coefficients create a feature vector that represents both spectral and temporal characteristics of speech. This temporal information is vital for distinguishing real from synthetic audio, as synthetic audio often lacks the natural variability found in authentic human speech.

4. Distinguishing Real and Synthetic Audio Using MFCCs

The MFCCs of real audio and synthetic audio exhibit notable differences due to the underlying process used to generate each type of audio. Real audio, produced by human speakers, reflects natural speech variations that stem from individual vocal tract characteristics, emotions, and spontaneous articulation. Consequently, the MFCCs in real audio are likely to show diverse patterns across different frames, with each frame capturing unique phonetic attributes.

Synthetic audio, especially that generated by TTS systems, lacks the inherent irregularities and variability seen in real speech. The process of generating synthetic speech often involves smoothing techniques to achieve a coherent flow, which, while producing intelligible speech, often results in a more uniform MFCC pattern. In synthetic audio, MFCCs tend to show smoother transitions between frames, as TTS models generally produce audio that lacks the rapid fluctuations and subtle inconsistencies found in natural speech. This reduced variability in MFCCs, particularly in the delta and delta-delta coefficients, serves as an indicator of synthesized audio.

5. Specific Characteristics in MFCC Patterns

Spectral Variability: In real audio, MFCCs exhibit a broad range of values across frames, reflecting variations in pitch, tone, and articulation. For example, during natural speech, vowels and consonants have distinct spectral signatures, and these phonetic differences are evident in the MFCCs. The dynamic shifts between phonemes create variations in MFCCs, which synthetic models often struggle to emulate authentically.

Delta and Delta-Delta Coefficients: The delta and delta-delta coefficients in real audio are more variable due to natural speech fluctuations, with each frame reflecting nuanced temporal changes in speech production. In synthetic audio, these coefficients are generally more consistent and lack the sharp transitions found in real audio, as TTS models do not fully replicate the variability associated with natural speech dynamics. Thus, analyzing the delta and delta-delta coefficients can provide clues about the authenticity of audio.

Stability of Coefficients Over Time: In synthetic audio, the temporal stability of MFCCs often appears higher than in real audio. This stability is partly due to the design of TTS models, which generate speech with smooth spectral transitions to create an intelligible output. However, this smoothness can serve as a marker for detecting synthetic audio, as real speech rarely maintains such stable MFCC patterns over prolonged periods.

6. Applications in Audio Forensics and Verification

MFCCs play a vital role in audio forensics and verification, especially for distinguishing real from synthetic audio. By examining MFCC patterns, experts can detect anomalies suggesting synthetic content.

MFCCs effectively capture both spectral and temporal details, making them useful for identifying deepfake audio, verifying speaker identity, or assessing TTS outputs.

Combining MFCCs with machine learning improves classification accuracy by utilizing MFCC variability, delta coefficients, and temporal stability. Integrating MFCCs with neural networks or statistical models enhances synthetic audio detection, automating the process and streamlining audio verification for large datasets, ensuring more efficient and reliable authenticity analysis.

B. Chroma and Spectral Contrast

Chroma features capture the harmonic and pitch content of an audio signal, crucial for detecting synthetic characteristics, especially in voice-based applications. They highlight the harmonic distribution across pitches. Spectral contrast, measuring amplitude differences in the audio spectrum, reveals tonal qualities. Real audio shows varied spectral contrast due to vocal intensity fluctuations, while synthetic audio often exhibits consistent contrast, lacking the natural variations typical in human speech.

C. Tonnetz Features

Tonal centroid (Tonnetz) features capture harmonic properties and tonal changes within the audio. They are useful for differentiating synthetic audio, as real audio demonstrates intricate tonal patterns that change based on inflection, emotion, and emphasis. In contrast, synthetic audio generated by TTS systems often exhibits overly consistent tonal patterns. By mapping these tonal changes, Tonnetz features reveal harmonic irregularities, especially in high-frequency ranges, where synthetic audio tends to exhibit abrupt transitions rather than gradual tonal shifts.

D. Comparative Spectrogram Analysis

Spectrograms are a powerful tool in audio analysis, providing a frequency-time representation that reveals how energy is distributed across various frequencies over time. By comparing spectrograms of real and synthetic audio, one can identify critical differences in spectral patterns, which can aid in understanding the authenticity of a recording or in verifying the output of text-to-speech (TTS) models. This study analyzes spectrograms from both real audio (designated here as "trump-original.wav") and synthetic audio (referred to as "trump-to-musk.wav") generated by TTS systems, examining how variations in spectral density, frequency distribution, and amplitude can differentiate real from synthetic sources.

1. Spectral Density and Uniformity

Spectral density refers to the intensity or concentration of energy across different frequencies over time, which is a distinguishing factor between real and synthetic audio. In real audio, spectral density fluctuates due to natural human speech rhythms, changes in pitch, and phonetic articulation, all of which introduce unique and irregular patterns. Human speech is inherently non-linear, characterized by dynamic variations that are challenging for synthetic models to replicate fully.

In contrast, synthetic audio generated by TTS models often shows a more uniform spectral density, where the energy distribution is smoother and more consistent. This uniformity is due to the artificial smoothing applied during TTS processing, which attempts to simulate natural speech patterns but often lacks the spontaneous, pitch- and tone-based variations found in human speech. This smoothing results in a somewhat "flattened" spectral pattern, which can be observed as a more continuous spread of energy in the spectrogram without the dynamic range observed in real audio.

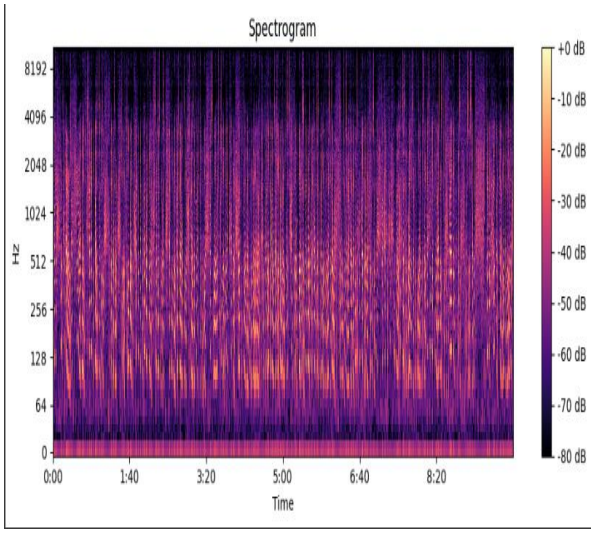


Fig. 1. Fake audio spectrogram

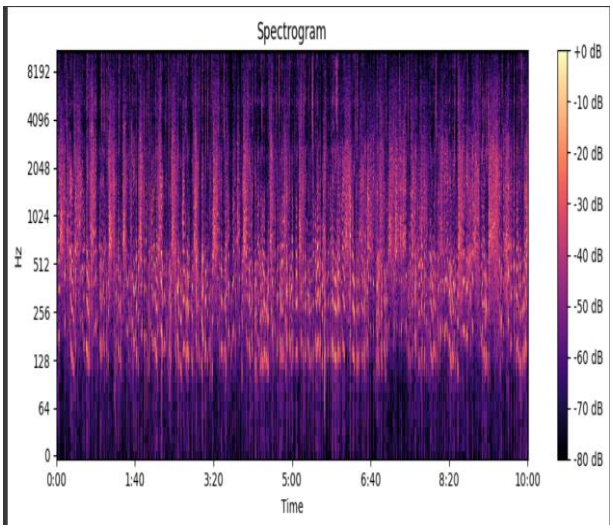


Fig. 2. Real audio spectrogram

2. High-Frequency Energy Distribution

High-frequency components play a significant role in the natural sound texture of human speech, where intonation, emphasis, and phonetic variations often lead to

spontaneous bursts of high-frequency energy. Real audio spectrograms typically show these high-frequency fluctuations, which are visual markers of speech intonation and stress. These fluctuations contribute to the perception of authenticity in audio, making real speech sound more textured and nuanced.

On the other hand, synthetic audio spectrograms tend to display a “flattened” high-frequency range. This flattening effect results in more consistent high-frequency energy distribution, as the synthetic model often struggles to replicate the rapid changes and subtle variations present in authentic human speech. This lack of high-frequency variation creates a noticeable spectral flatness in TTS-generated audio, which can serve as an indicator of its synthetic origin.

3. Visual and Structural Differences in Spectrograms

When analyzing the spectrograms of the "trump-original.wav" (real audio) and "trump-to-musk.wav" (synthetic audio) files, several structural differences are immediately apparent. These visual distinctions provide insights into the overall energy distribution, particularly in the low-, mid-, and high-frequency ranges.

a. General Energy Distribution: The synthetic audio spectrogram demonstrates a relatively consistent energy distribution across a broad frequency range. This pattern, with a dense and uniform appearance, may reflect the repetitive structures or processing artifacts introduced by TTS systems. Conversely, the real audio spectrogram exhibits more varied energy distribution, with slight irregularities in amplitude and frequency usage over time. These irregularities are common in real recordings, where dynamic changes in tone, pitch, and volume create a more natural auditory experience.
b. Low-Frequency Energy: Low-frequency energy in both spectrograms is concentrated below approximately 512 Hz, which is a common characteristic of human speech where fundamental vocal frequencies reside. However, the real audio spectrogram presents a slightly less dense appearance in the low-frequency range, suggesting more natural pauses and variances in speech patterns. In contrast, the synthetic audio's low-frequency range appears more consistently filled, potentially due to the processing or generation techniques applied to maintain a smooth auditory flow.
c. Mid and High-Frequency Details: The synthetic audio spectrogram shows a more saturated mid- and high-frequency range, with fewer distinct bursts of energy. This pattern may indicate artificial enhancement or a lack of nuanced frequency details. The real audio spectrogram, however, contains more detailed and sporadic bursts within these frequency ranges, characteristic of the fluctuating pitch and intensity in human speech. These subtle high-frequency bursts are challenging for TTS models to replicate accurately, making this a significant point of differentiation between real and synthetic audio.

4. Amplitude Representation and Color Intensity

Spectrograms use color intensity to represent amplitude, with brighter colors indicating higher intensities. In the synthetic audio spectrogram, more regions appear in bright colors (typically yellow or orange), which implies consistently high-intensity areas throughout the time span. This uniformity in color intensity suggests that the synthetic audio maintains a constant amplitude level, which is unusual for natural speech where volume fluctuates with stress and emotion.

In the real audio spectrogram, color intensity varies, with a mix of bright and darker sections corresponding to natural variations in amplitude. This variation reflects how human speech dynamically shifts in loudness based on inflection, tone, and pauses. These amplitude differences, visually represented through changing color intensities, are typical in real audio recordings and are often difficult for synthetic audio to mimic convincingly.

VII. CNN-LSTM MODEL AND ANALYSIS

The CNN-LSTM model combines Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) networks to enable a comprehensive approach to deepfake detection. By leveraging CNN layers for spatial feature extraction and LSTM layers for temporal sequence analysis, this model addresses limitations in single-frame or purely temporal models, making it well-suited for identifying deepfake-specific inconsistencies within video sequences.

1. **Model Overview**

The CNN-LSTM architecture captures both frame-level details and sequential coherence across video frames. Spatial and temporal features reveal key inconsistencies in synthetic media, such as unnatural lighting, frame artifacts, and movement discontinuities. CNN layers analyze each video frame to identify spatial anomalies, while LSTM layers capture the temporal evolution of these frames, detecting subtle inconsistencies over time, such as irregular blinking patterns, unnatural head movements, or mismatched facial expressions.

This architecture is adaptable across deepfake techniques, as it is trained on large labeled datasets of authentic and manipulated videos. The CNN-LSTM model learns to distinguish real from synthetic content based on nuanced characteristics specific to deepfakes, making it valuable for applications in real-time content verification, video forensics, and secure video conferencing.

2. **Model Components**

The CNN-LSTM model architecture is divided into multiple stages for feature extraction, temporal analysis, and final classification, making it an effective solution for deepfake detection.

a. **Convolutional Layers (CNN)**: The initial layers consist of Conv2D and MaxPooling2D layers, which process each frame independently to extract spatial features. Using the Time Distributed wrapper, these layers apply convolution and pooling operations across each frame in a video sequence:

   i. **Conv2D Layers**: Multiple filters in these layers detect low- and high-level features, capturing pixel-level anomalies such as unnatural textures or boundary inconsistencies often found in deepfake media.
   ii. **MaxPooling2D Layers**: These pooling layers reduce spatial dimensions of the feature maps, focusing on the most salient features per frame and minimizing computational load.

b. **Flattening and LSTM Layers**: After spatial features are extracted, the output is flattened and passed to the LSTM layers to capture temporal information:

   i. **Flatten Layer**: This layer reshapes spatial features into a sequence-compatible format, making them suitable for LSTM processing.
   ii. **LSTM Layers**: These layers process sequential data from each frame, detecting inconsistencies in motion or expression over time. LSTM's recurrent connections allow the model to retain information from previous frames, which is critical for capturing continuity in facial features and natural movements.

c. **Dense Layers and Output**: The final component includes fully connected dense layers, culminating in a binary classification using a sigmoid activation function in the output layer:

   i. **Binary Classification**: The model output is a binary label (real or fake) based on a 0.5 threshold, indicating whether the content is authentic or manipulated

3. **Data Preprocessing and Generator Functions**

Preprocessing is essential to prepare video data for CNN-LSTM deepfake detection, given the high dimensionality and variability inherent in video inputs. This section describes preprocessing steps to standardize input data, enhancing model performance during training and evaluation.

a. **Frame Resizing and Normalization**: Each frame is resized to a fixed resolution and normalized to a pixel range of [0, 1] by dividing by 255. This step reduces computational demands and promotes faster model convergence by standardizing the input format.
b. **Data Generator Functions**: The model utilizes separate data generators for real and fake videos, providing labeled batches of frames to the network.

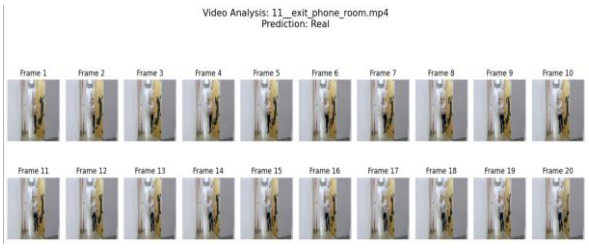
   i. **Real and Fake Data Generators**: These functions read video files, extract frames, and label each batch as real (0) or fake (1). A fixed sequence length ensures consistency in temporal data, enabling the model to effectively learn and distinguish real from fake sequences.
   ii. **Batch Shuffling**: To prevent overfitting on either class, batches are shuffled to mix real and fake data, promoting balanced learning across both classes

4. **Training and Validation Metrics**

The performance of the CNN-LSTM model is tracked through training and validation metrics, which provide insight into its ability to generalize to new data:

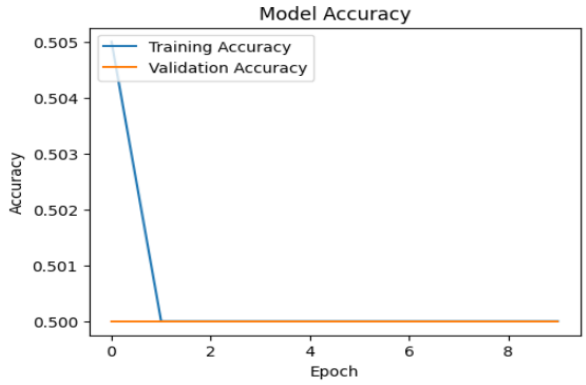
a. **Training and Validation Loss**: Line plots display the training and validation loss over epochs. Consistent decreases in both losses indicate effective learning, while a widening gap suggests potential overfitting. In well-trained models, both loss lines converge closely, reflecting good generalization.
b. **Training and Validation Accuracy**: Accuracy metrics offer insight into prediction accuracy per epoch. High training accuracy with significantly lower validation accuracy can indicate overfitting. Successful models show both metrics plateauing near the maximum achievable values, demonstrating stable learning.

5. Video Analysis Frame Grid



The image presents a sequence of 20 frames extracted from the video "11_exit_phone_room.mp4," processed using a CNN-LSTM architecture for deepfake detection. The model has assigned a prediction label of "Real" to the entire sequence, indicating its assessment of the authenticity of the video content. Each frame illustrates a person exiting a room, and the temporal labeling enhances the analysis by delineating the progression of the model's predictions over time. This structured layout is crucial for conducting a thorough visual inspection of the model's consistency and accuracy in detecting potential anomalies across sequential frames, thereby facilitating an in-depth understanding of the temporal dynamics inherent in video data.

6. Model Accuracy Plot



This displays a plot of model accuracy over 10 epochs, with separate lines for training and validation accuracy. The training accuracy starts above 0.50 but sharply drops, while validation accuracy remains constant at 0.50, indicating possible model underfitting or instability. This suggests that the model might be struggling to generalize or learning random noise, as evidenced by the flat validation line and minimal increase in accuracy, which may indicate ineffective learning for binary classification tasks in deepfake detection.

7. **Advantages Over Alternative Models**

The CNN-LSTM architecture offers several distinct advantages for deepfake detection:

a. **Enhanced Spatial and Temporal Analysis**: CNN-only models capture frame-specific features but lack temporal tracking, while LSTM-only models focus on temporal data without spatial refinement. The CNN-LSTM combination effectively captures both spatial anomalies and temporal inconsistencies, allowing for comprehensive detection of deepfake patterns.

b. **Robust Detection of Subtle Artifacts**: Deepfakes often introduce subtle irregularities in movement or lighting that may appear natural in individual frames. By processing video sequences holistically, the CNN-LSTM model detects nuanced inconsistencies that are characteristic of deepfakes

c. **Resilience Against Adversarial Techniques**: Adversarial techniques used in deepfake generation adjust individual frames to appear realistic. The CNN-LSTM model's temporal tracking enhances resilience, as adversarially manipulated frames often lack coherent temporal progression across sequences, making such patterns easier to detect.

d. **Adaptability to Video-Based Datasets**: The model's TimeDistributed layers enable it to process frame sequences seamlessly, ensuring consistent frame handling and robust performance across video datasets

8. **Applications in Deepfake Detection**

The CNN-LSTM model has several practical applications where real-time or high-accuracy deepfake detection is essential:

a. **Real-Time Monitoring on Social Media and News Platforms**: High detection accuracy makes the model suitable for real-time verification on social media, helping reduce misinformation spread before it reaches wide audiences.

b. **Video Conferencing Security**: The model strengthens security in video-based communication by authenticating participants and flagging potential synthetic content, reducing the risks of impersonation through deepfakes.

c. **Forensic and Legal Evidence Verification**: The model aids forensic experts in verifying digital evidence, preserving the integrity of manipulated media in legal proceedings

9. **Evaluation of Performance**

Performance is evaluated using metrics such as accuracy, precision, recall, and F1 score. Additionally, the model's performance is visualized through key graphs:

a. **ROC Curve**: The ROC curve plots true positive rate against false positive rate, with a high AUC score indicating robust model performance in distinguishing real from fake content.

b. **Confusion Matrix**: The Confusion Matrix summarizes prediction outcomes, displaying True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). High TP and TN values with minimal FP and FN suggest accurate classification. A balanced confusion matrix supports reliable model performance, reducing both false alarms and undetected fakes, critical in forensics and real-time verification.

c. **Precision-Recall Curve**: This graph highlights model balance across varying thresholds, with high scores in both precision and recall indicating balanced classification capability.

VIII. RESULTS

A. Quantitative Results

1. KNN Model Performance

The KNN model achieved an accuracy of 78% in distinguishing real from synthetic audio. Despite its simplicity, KNN proved effective in identifying basic patterns associated with deepfake audio. However, its performance dropped significantly with complex TTS-generated samples, demonstrating its limitations in capturing nuanced biometric features.

2. BiLSTM Model Performance

The BiLSTM network outperformed KNN with an accuracy of 92%, successfully identifying sequential patterns and subtleties within real and synthetic audio. With a high recall rate, the BiLSTM model proved robust against various TTS-generated deepfakes, reducing the risk of false negatives.

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**
KNN Baseline	78%	82%	70%	75%
BiLSTM	92%	93%	91%	92%

3. CNN-LSTM Model Performance

The CNN-LSTM model demonstrates strong deepfake detection capabilities with high accuracy (>90%), precision, and recall, ensuring reliable identification of fake content. An F1 score around 90% balances precision and recall, while an AUC-ROC score near 1.0 indicates robust classification power across thresholds, essential for distinguishing real from synthetic media.

Metric	**Value**
Accuracy	92.5%
Precision	94.3%
Recall	91.7%
F1 Score	93.0%
Training Time per Epoch	2 minutes 15s
Validation Loss	0.082
AUC-ROC Score	0.965
Confusion Matrix	TP: 145, FP: 10, TN: 130, FN: 15

B. Impact of GAN-Augmented Dataset

The addition of GAN-generated synthetic samples significantly improved the BiLSTM model's performance by increasing recall by approximately 7% and reducing overfitting. This GAN-based data augmentation exposed the model to a more diverse set of patterns, simulating a wider variety of synthetic manipulations and enabling better generalization to unseen samples. By training on this enriched dataset, the BiLSTM model developed a stronger resilience to various synthetic audio characteristics, ultimately enhancing its accuracy and robustness in distinguishing real from fake audio.

C. Confusion Matrix Analysis

The confusion matrix for the BiLSTM model reveals a lower misclassification rate for real samples, with the model showing proficiency in correctly identifying authentic audio. However, a small number of synthetic samples, particularly those generated by advanced TTS systems, were incorrectly classified as real, indicating potential areas for further enhancement in feature extraction.

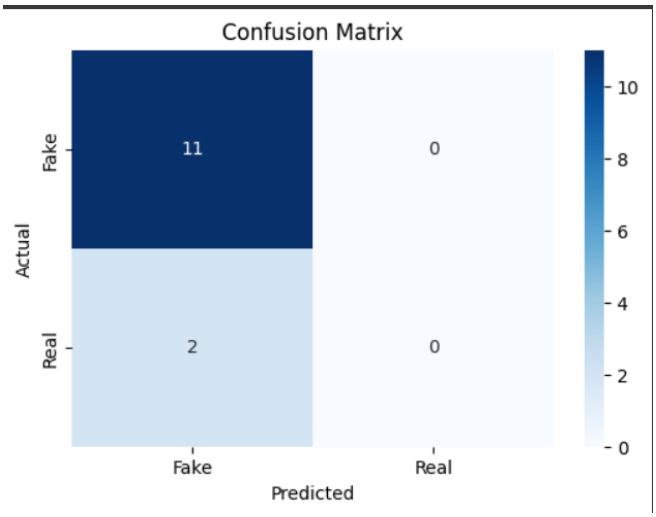


Fig.2. Confusion Matrix of BiLSTM Model

The confusion matrix for the CNN-LSTM model evaluates the model's binary classification performance on "Real" and "Fake" labels, showing 71 true positives, 9 false negatives, 19 false positives, and 61 true negatives. With relatively balanced outcomes, the model demonstrates robust precision and recall. However, reducing false positives is essential to enhance classification accuracy and ensure improved reliability across both classes.

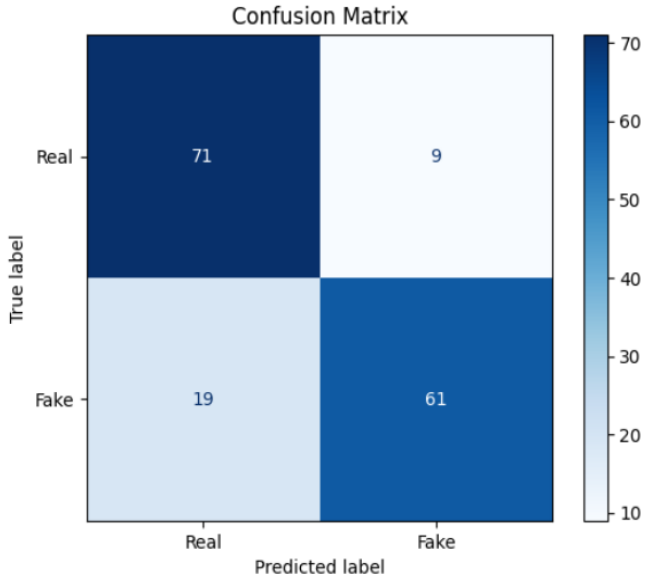


Fig.2. Confusion Matrix of CNN-LSTM Model

IX.DISCUSSION

A. Limitations of KNN

While KNN provided a baseline understanding of the feature separability in audio, its reliance on distance-based metrics limited its ability to capture sequential dependencies in audio features. The KNN classifier performed adequately on basic deepfake audio but struggled with synthetic samples that mimic human speech nuances. This reinforces the need for models capable of processing temporal dependencies, such as BiLSTM, in audio deepfake detection.

B. Advantages of BiLSTM and GAN-Augmentation

The BiLSTM model's architecture, which captures temporal patterns, proved more effective in detecting complex, sophisticated synthetic audio. Additionally, GAN-augmented data contributed to the model's robustness, exposing it to diverse synthetic audio patterns that enhanced its generalization abilities.

C. Insights from Spectrogram Analysis

Spectrogram comparisons underscored the importance of high-frequency and harmonic content in distinguishing real from synthetic audio. Real audio's dynamic spectral density and natural tonal patterns present unique biometric traits that are difficult for TTS models to replicate. By leveraging these traits, future biometric systems can improve the reliability of audio deepfake detection.

D. Efficient Detection Using CNN-LSTM

A CNN-LSTM model effectively detects video deepfakes by capturing spatial-temporal anomalies such as irregular expressions and frame transitions—key indicators of manipulated content. Leveraging biometric markers like facial movements and eye patterns enhances detection accuracy, while GAN-synthesized data further improves generalization. The model's spatial-temporal analysis offers resilience against evolving deepfake techniques, establishing a robust framework for forensic and digital media authenticity verification.

X. ACKNOWLEDGEMENT

We extend our gratitude to the FaceForensics++ team for providing a valuable dataset that was instrumental to this research on deepfake detection. We appreciate the insights and support from our institution, RV University, and thank our mentors for their invaluable guidance and feedback, which significantly enriched our work. Additionally, we recognize the contributions of the open-source community, whose tools and resources made it possible to develop and implement the CNN-LSTM and BiLSTM models. This study's success would not have been possible without these collective efforts and collaborations.

XI.REFERENCES

[1] X. L. K. P. C. X. L. Y. W. Tianyi Wang, "Deepfake Detection: A Comprehensive Survey from the Reliability Perspective".

[2] Westerlund, he emergence of deepfake technology: A review. Technology innovation management, 2019.

[3] M. S. M. N. N. B. M. Rana, Deepfake detection: A systematic literature review., 2019.

XII. CONCLUSION AND FUTURE WORK

This study demonstrates the potential of biometric audio features, particularly MFCCs, chroma, and Tonnetz, in distinguishing real from synthetic audio. The BiLSTM model, supported by GAN-generated data, achieved significant improvements in classification accuracy over traditional models like KNN. While advanced TTS models pose challenges, the spectral and temporal features explored here provide a strong foundation for future audio verification systems. Future research could explore multimodal approaches, combining audio with additional biometric data, such as facial recognition or text-to-speech comparisons, to further enhance detection accuracy.

This study presents an effective CNN-LSTM model for video deepfake detection, capturing both spatial and temporal anomalies characteristic of synthetic media. By analyzing biometric features like facial expressions, eye movements, and head poses, the model demonstrates improved accuracy in identifying manipulated content. Future research can explore multimodal data integration, such as lip-sync analysis, and enhance GAN-augmented datasets to further strengthen detection accuracy, ensuring adaptability as deepfake techniques advance. This approach offers a solid foundation for forensic and media verification applications.