Detecting Audio Deepfakes using Deep Learning

Speech Project Proposal

Submitted by:

102217080 – Shivansh Verma

102217069 – Divyanshu Kumar

102206183 – Gargi Mehta

102217015 - Tanisha

102217061 – Bhavneet Kaur

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Thapar Institute of Engineering and Technology, Patiala

May, 2025

TABLE OF CONTENTS

Abstract

With the rapid advancement of AI-generated speech technologies like Text-to-Speech (TTS) and Voice Conversion (VC), it's becoming increasingly difficult to distinguish real human voices from fake ones. This project tackles the challenge of detecting such deepfake audio using a deep learning approach that combines signal processing with convolutional neural networks (CNNs). We use Mel-spectrograms—a visual representation of sound frequencies over time—as the core input feature, capturing the unique acoustic patterns of both genuine and synthetic speech. The model is trained on the ASVspoof2021 Logical Access dataset and tested against real-world examples from a Kaggle dataset containing deepfake voice recordings. A custom-built 3-layer CNN processes the spectrograms to classify each sample as either "bonafide" or "spoof." Despite the model's relatively simple architecture, it demonstrates strong accuracy in identifying fake audio, even when tested on unseen data. Our results show that combining Mel-spectrogram analysis with lightweight CNNs is not only effective but also practical for real-world applications where speed and reliability are essential.

Introduction

1. Understanding Audio Deepfakes

Audio deepfakes are artificially generated or manipulated audio recordings that mimic human speech with high fidelity. Leveraging advancements in deep learning, particularly in generative models like GANs and autoencoders. Audio deepfakes are computer-generated or heavily edited voice recordings that are designed to sound like a real human speaking. These recordings are not just imitations — they can be made to sound almost indistinguishable from an actual person's voice, including their accent, tone, pitch, and speaking style. Such capabilities pose significant threats, including misinformation dissemination, identity theft, and fraudulent activities .

2. Dangers of Audio Deepfakes

- **Misinformation**: A fake voice recording of a public figure (e.g., a politician or celebrity) can be created and shared online to spread false information.
- **Fraud and Scams**: Attackers could pretend to be someone's boss or family member over the phone and trick them into sending money or sharing confidential information.
- **Identity Theft**: By mimicking someone's voice, attackers might bypass voice-based authentication systems used by banks or security services.
- **Blackmail and Extortion**: People could be falsely represented saying inappropriate or illegal things, damaging reputations and careers.

3. The Imperative for Detection

As the accessibility of tools to create audio deepfakes increases, so does the potential for misuse. Detecting these manipulations is crucial to maintain trust in digital communications, protect individuals' identities, and ensure the integrity of information. Traditional methods often fall short due to the sophistication of modern deepfake techniques, necessitating advanced detection mechanism.

4. Overview of the Project

In this project, the aim is to automatically detect audio deepfakes — voice recordings that have been artificially created or altered to sound like real people. Detecting such manipulations is difficult because the fake audio can sound very convincing to the human ear. That's why this project uses deep learning techniques, specifically a Convolutional Neural Network (CNN)**,** to automate the detection process with high accuracy.

The system takes an audio file and determines whether it is:

- **Bonafide** (genuine, unaltered human speech), or
- **Spoof** (manipulated or synthetic speech, created by algorithms

Literature Survey

1. Key Concepts and Definitions

1.1. Deepfake Audio / Speech Spoofing

Synthetic audio attacks include Text-to-Speech (TTS) and Voice Conversion (VC) systems that generate fake speech. TTS models convert text to speech (e.g. WaveNet, Tacotron) while VC alters one speaker's voice to sound like another. These deepfake voices can deceive speaker verification systems and spread misinformation. The ASVspoof challenge series (2015–2023) provides standardized corpora (Logical Access, Physical Access, DeepFake tracks) to benchmark countermeasure (CM) systems.

1.2. Spectrogram and Mel-Spectrogram

A spectrogram is a time-frequency image of audio. The Mel spectrogram applies a perceptual Mel-frequency scale, often using log amplitudes, to capture speech characteristics. Converting audio to a Mel spectrogram is common in deepfake detection, since it encodes spectral and temporal patterns effectively. For example, Kashif et al. use log-Mel spectrograms as CNN input; Luo & Sivasundari similarly preprocess raw waveforms into log-Mel spectrograms before CNN processing. Other features include MFCCs, CQCCs, and LFCCs, but they may omit some artifacts. Gao et al. found that novel 2D DCT spectro-temporal features (over log-Mel) better capture deepfake artifacts than MFCC/CQCC.

1.3. Convolutional Neural Networks(CNNs)

CNNs treat spectrograms as images, learning hierarchical filters to distinguish real vs. spoofed speech. They are effective for capturing local spectral patterns. Other neural architectures include RNNs/LSTMs (modeling temporal context), CRNNs (CNN followed by RNN), and more recent Transformer-based encoders or graph networks (e.g., AASIST).

Metric terms**:** Equal Error Rate (EER) and t-DCF measure CM performance.

2. Pros and Cons of Different Approaches

2.1. CNN on Mel-Spectrograms

Many studies confirm CNNs on Mel-spectrogram inputs yield low EER. Kashif et al. achieved robust spoof detection using a ResNet-34 CNN on Mel-spectrograms. Bajwa et al. evaluated multiple CNN backbones (VGG, ResNet, DenseNet, etc.) on Mel-spectrograms and found high accuracy and robustness, even under added noise. **Pros:** Captures local spectral artifacts; leverages powerful vision CNNs. **Cons:** May overfit to known data; computational cost for deep CNNs. Xception-style CNNs have very fine-grained filters but can latch onto identity cues.

2.2. Handcrafted Features + Classical Models

Traditional features (MFCC, CQCC, LFCC) fed to GMM/SVM were strong baselines (ASVspoof 2019 used CQCC+GMM). However, Gao et al. showed even shallow CNNs outperform GMM/SVM by exploiting spectrogram anomalies. **Pros:**Simple,interpretable.
Cons: Limited generalization to unseen attacks.

2.3. Deep Time-Domain Models

End-to-end models (sinc-convolution networks, RawNet, wav2vec embeddings) process raw waveforms. For example, AASIST integrates sinc-conv and RawNet2 to operate on raw audio. Shaaban & Yildirim used a Siamese CNN on raw waveforms with a custom contrastive loss, achieving low EER (2.95%) on ASVspoof 2019. **Pros:** No feature engineering, can learn subtle waveform artifacts. **Cons:** Requires more data/training; sensitive to channel variation.

2.4. Hybrid and Ensemble Systems

Recent methods fuse features or models. Neelima & Prabha combined MFCC, CQCC, and spectral features in a CNN+LSTM ensemble, reporting high accuracy. Pham et al.

fused CNNs, RNNs, and pretrained audio encoders over multiple spectrogram types, achieving SOTA (EER ≈ 0.03%) on ASVspoof 2019.
Pros: Improved robustness by capturing diverse cues.
Cons: High complexity; risk of overfitting without careful regularization.

2.5. Self-Supervised and Pretrained Models

Leveraging ASR or SSL models as feature extractors (e.g. wav2vec, Whisper) has yielded gains. Luo & Sivasundari fine-tune the Whisper transformer on spoof detection, integrating AASIST, to achieve state-of-art EER (2.85%).
Pros: Strong general features.
Cons: Require large pretrained backbones and computational resources.

3. Relevant Theories

- Deepfake detection leverages representation learning: spectrograms embed speech as 2D signals, so 2D CNN theory applies (convolutions capture local spectral-temporal patterns).
- Graph neural networks (e.g. AASIST) apply graph-theoretic processing on spectrogram representations.
- Attention and Transformer theory underpins models like Whisper that use self-attention layers to capture long-range context.
- Contrastive learning principles appear in Siamese CNNs (Shaaban) where pairs of bonafide vs. spoof are compared, optimizing a contrastive loss to cluster genuine audio.
- Adversarial robustness and meta-learning (e.g., Wang & Hansen) are emerging: they employ adversarial augmentation and meta-losses to improve generalization.
- Classical signal processing theory underlies feature engineering: for instance, Gao et al. use 2D Discrete Cosine Transform on log-Mel spectrograms to capture long-range modulation features—a concept from audio signal analysis.

4. Similarities and Differences Across Studies

- Many studies converge on Mel-spectrogram + CNN as an effective pipeline. For example, Kashif et al. and Bajwa et al. both report high accuracy with simple CNNs on Mel-spectrograms.
- Others augment this with deeper or residual networks (ResNet, Xception).
- A common feature is the use of log compression and Mel filterbanks to emphasize human-audible frequencies.

4.1. Differences arise in:

- **Model complexity and fusion:**
  Pham et al. combined multiple spectrogram types (STFT, CQT, wavelet) and model ensembles, whereas Kashif et al. used a single ResNet on Mel-spectrogram.
- **Feature types:** Gao et al. explore 2D-DCT as a drop-in replacement for Mel/MFCC.
- **Datasets:** Most use ASVspoof (2015–2021) or ADD challenge data; some use real-world "in-the-wild" corpora.
- **Metrics:** Some use EER, others use min t-DCF.

5. Critical Analysis of Techniques

5.1. CNN on Mel-spectrograms

- Strength: Simple and effective on known spoof types.
- Risk: Overfitting. Xception/CNN models may learn speaker/channel cues—hurting generalization.
- High in-domain accuracy may not generalize well to unseen datasets.

5.2. Feature Engineering

- Traditional features (MFCC, LFCC, CQCC) capture broad info but may miss deepfake artifacts.
- Empirically, Mel spectrograms often outperform MFCC.
- Gao et al**.** added spectro-temporal modulation features (e.g., 2D-DCT) → improved detection.
- Limitations: Handcrafted features need domain insight and can miss artifacts.

5.3. Raw Waveform Models

- Capture phase and waveform nuances lost in spectrograms.
- Drawbacks: Data-hungry, sensitive to recording conditions.
- Examples:
    - Whisper + AASIST: EER = 2.85%, but requires large pretrained models.
    - Shaaban (Siamese CNN**):** Achieved competitive results on raw waveforms.

5.4. Ensembles and Transfer Learning

- Best results often come from combining feature types/models (Pham et al.).
- Cons: Complexity, lower interpretability.
- Transfer learning from pretrained CNNs (e.g., ResNet, Whisper) introduces potential for learning irrelevant biases.
- Need careful tuning; computation-heavy.

5.5. Conclusion:

CNN + Mel pipelines are strong, simple baselines. Emerging methods (e.g., attention, graph networks, contrastive learning) may outperform them, but at higher complexity.

6. Connection Between Literature and the Current Project

- This project uses TensorFlow + Librosa to extract 5-second Mel-spectrograms.

- Trains a 3-layer **CNN** to classify bonafide vs. deepfake audio.
- Aligned with:
    - Kashif et al.: Used a deeper ResNet on Mel-spectrograms.
    - Bajwa et al.: Used Mel-spectrogram images and CNN backbones.
- Training data from ASVspoof 2021 LA—a common benchmark.
- Test set: Kaggle “DEEP-VOICE”, real-world data akin to "in-the-wild" corpora.
- Key differences**:**
    - Simpler model (3-layer CNN vs ResNet/Xception).
    - No use of temporal models (LSTM) or transformer encodings.
    - Uses 5-sec clips, similar to Whisper+AASIST (4–30 sec inputs).

7. Summary of Key Findings

7.1. Feature Representations

- Mel-spectrograms are widely validated and often outperform MFCC/CQCC.
- 2D-DCT over Mel spectrograms can boost detection.
- Raw waveform models (e.g., sinc-CNN, RawNet) also achieve strong results.

7.2. Model Architectures

- CNN-based classifiers are common and effective.
- Simple CNNs on spectrograms yield low EER.
- Advanced CNNs (ResNet, Xception) with residuals are popular.
- RNN/LSTM add temporal modeling.
- Graph-based (AASIST) and Transformer encoders are emerging.
- Ensembles/multi-stream models often yield the best performance.

7.3. Dataset and Tasks

- Most work focuses on ASVspoof (2015–2021) and ADD challenges.
- Logical Access (LA) is the typical deepfake detection scenario.
- Few address “in-the-wild” or uncontrolled real-world data.

7.4. Performance

- Top methods report:
    - EER ~0.03% on ASVspoof 2019 (Pham et al.).
    - ~3% on ASVspoof 2022 ADD.
- Siamese CNNs with contrastive loss: EER ~2.95%.
- Simpler CNNs still get <5% EER on known datasets.

7.5. Generalization

- Trade-off between complexity and generalization.
- Models trained on ASVspoof often fail on unseen data or new attack types.

8. Identified Gaps in Literature

8.1. Generalizability:

- Identity leakage and overfitting are common.
- Weak performance on unseen speakers or in-the-wild samples.

8.2. Adversarial Robustness:

- Few works simulate adversarial attacks or strong perturbations.
- Some like Wang & Hansen show gains via adversarial augmentation.

8.3. Explainability:

- Some use XAI (e.g., Grad-CAM), but field lacks deep insight into model behavior.

8.4. Short vs Long Input Handling:

- Most studies use fixed-length (e.g., 4–30 sec) clips.
- Real attacks may include very short phrases.

8.5. Heterogeneous Data:

- Many methods specialize in TTS or VC.
- Few detect mixed-source or switching attacks.

8.6. Resource Efficiency:

- SOTA models are heavy (transformers, ensembles).
- Few approaches are lightweight enough for real-time/edge devices.

9. Recommendations for Future Work

- **Cross-Dataset Evaluation:**Test models on diverse corpora (ASVspoof + Kaggle/in-the-wild).
- **Data Augmentation & Adversarial Training**:Use perturbations (noise, reverberation, adversarial samples).
- **Model Innovations:**Try CNN+Transformer hybrids, metric learning (e.g., Siamese/triplet loss).
- **Feature Fusion:**Combine handcrafted and learned features (e.g., Mel + residuals + modulations).
- **Explainable AI:**Systematically apply XAI to guide feature and architecture design.
- **Efficiency:**Design lightweight countermeasures (3-layer CNNs, model pruning).
- **New Datasets:**Develop large, realistic deepfake speech datasets (cross-lingual, noisy).

10. Tables

10.1. Table 1:Model architectures and performance

This table summarizes representative models from the literature, their input features, datasets, and reported performance (EER or accuracy). Models vary from simple CNNs on spectrograms to complex ensembles or pretrained networks.

Table 1: Anti-Spoofing Model Comparison Table

Model / Method	Input Features	Dataset(s)	Performance (Eval Metric)
ResNet-34 CNN	Log-Mel spectrogram (244 × 244)	ASVspoof 2019 LA	EER ≈ low (ResNet → state-of-art for single model)
CNN-LSTM (Hybrid)	Combined MFCC, CQCC, spectral (236×)	ASVspoof 2019 LA	High accuracy (“high accuracy” in freq+time)
Siamese CNN	Raw waveform (2-channel pair)	ASVspoof 2019 LA	98% accuracy (EER ≈2.95%)
AASIST GraphCNN	Raw waveform (sinc-conv + RawNet2)	ASVspoof 2021 DF	EER ≈2.85% (ASVspoof 2021 DF)
Whisper + AASIST	Log-Mel spectrogram (Transformer)	ASVspoof 2021 DF	EER = 2.85%
Ensemble (CNN+RNN+SSL)	Mel/CQT/Wavelet spectrograms	ASVspoof 2019 LA	EER = 0.03% (top-3)
Multi-CNN (Grad-CAM)	Mel spectrogram	FakeAVCelebV2	High acc. (effective under noise)

10.2. Table 2: Common feature extraction techniques

Features capture different audio aspects. References indicate typical usage. Advantages/disadvantages are relative to spoof detection.

Table 2: Audio Feature Comparison Table

Feature	Type	Usage in Literature	Pros	Cons
Mel-Spectrogram	Time-frequency (244 Mel bins)	Widely used (baseline for CNN)	Captures perceptual spectral shape; simple to compute; proven effective	Loses phase; may miss fine temporal artifacts.
Log-Scaled Spectrogram	Linear-frequency (FFT)	Sometimes used (CNN input)	Full frequency resolution; no Mel warp	Larger size; less perceptually tuned.
MFCC / LFCC / CQCC	Cepstral coefficients	Traditional ASVspoof baselines	Compact; well-studied for speaker/ASV tasks	May smooth out generation artifacts.
2D-DCT over Mel	Spectro-temporal (proposed)	Gao et al. feature	Captures long-range modulations; improved artifact sensitivity	Novel; requires validation on varied data.
Raw Waveform	Time-domain samples	Used by AASIST, Siamese CNN	Potentially retains all info; end-to-end learning	Heavy computation; needs lots of data.

10.3. Table 3:Relevant datasets for deepfake audio

Datasets vary in scope. Many studies use ASVspoof challenges; Kaggle sets (e.g. DEEP-VOICE) are smaller and more recent.

Table 3 : Relevant Datasets for Deepfake Audio

Dataset / Corpus	Content	Spoof Type	Usage / Notes
ASVspoof 2019 LA	19,000 real + 165k spoof	TTS, VC (logical)	Widely used benchmark; contains diverse synthetic attacks.
ASVspoof 2021 LA	~25k real + 128k spoof	TTS, VC (with compression)	Emphasizes compression effects; current project's train set (ASVspoof2021.LA.cm.train).
ASVspoof 2021 DF	>60k real + 144k spoof	Deepsynthesis, VC	"In-the-wild" DF task (unknown attacks); tests robustness.
DEEP-VOICE (Kaggle)	Few minutes bona-fide vs. deepfake clips	TTS/VC-based	Real versus AI-generated speech; used in recent Kaggle competition.
FakeAVCelebV2	AI-generated voices (21k) vs celebrities	TTS/VC	Used by Bajwa et al. for CNN experiments.
WaveFake (Kaggle)	~1000 malicious vs genuine	Multi-TTS (WaveNet, etc.)	Kaggle audio deepfake detection challenge dataset.

Methodology

1. Dataset Utilization

The system employs the ASVspoof 2021 dataset, a benchmark dataset designed for evaluating spoofing countermeasures in automatic speaker verification. This dataset comprises a diverse collection of genuine and spoofed audio recordings, providing a robust foundation for training and testing the detection model.

2. Data Preprocessing

To prepare the audio data for CNN input, the following preprocessing steps are undertaken:

- Mel Spectrogram Conversion: Audio files are converted into Mel spectrograms, capturing the frequency domain representation of the audio signal. Before the audio can be fed into a deep learning model, it needs to be converted into a format that the model can understand**.** This is done using Mel spectrograms.

  What is a Mel spectrogram?

  A Mel spectrogram is a visual representation of sound, where:

- The x-axis represents time,
- The y-axis represents frequency (on a Mel scale, which mimics human hearing), and
- The color/intensity represents the energy or loudness of each frequency at each time.By converting audio to spectrograms, the model can analyze patterns in the frequency domain, which are often where deepfake manipulations leave subtle traces.
- Data Augmentation: Techniques such as noise addition, pitch alteration, and time-stretching are applied to enhance the diversity of the training data, improving the model's generalization capabilities.

3. Model Architecture

The CNN architecture is tailored to extract salient features from the Mel spectrograms:

- **Convolutional Layers**: Multiple convolutional layers capture local patterns in the spectrograms.
- **MaxPooling Layers**: These layers reduce the spatial dimensions, retaining essential features while minimizing computational load.
- **Batch Normalization**: Applied to stabilize and accelerate the training process.
- **ReLU Activation**: Introduces non-linearity, enabling the network to learn complex patterns.
- **Dropout Layers**: Implemented to prevent overfitting by randomly deactivating neurons during training.
- **Global Average Pooling**: Aggregates feature maps, reducing the data dimensionality before the final classification.
- **Dense Layer with Sigmoid Activation**: Produces the final binary classification output, indicating the likelihood of the audio being genuine or spoofed.

4. Training Process

The model is trained using the following parameters:

- **Loss Function**: Binary Cross-Entropy, suitable for binary classification tasks.
- **Optimizer**: Adam optimizer, chosen for its efficiency and adaptive learning rate capabilities.
- **Evaluation Metrics**: Accuracy, F1 Score, ROC Curve, and AUC are monitored to assess the model's performance.

5. Evaluation and Visualization

Post-training, the model's performance is evaluated on a separate test set. Visualization tools like plot_model and Netron are utilized to depict the model architecture, facilitating better understanding and potential debugging.

Work Flow

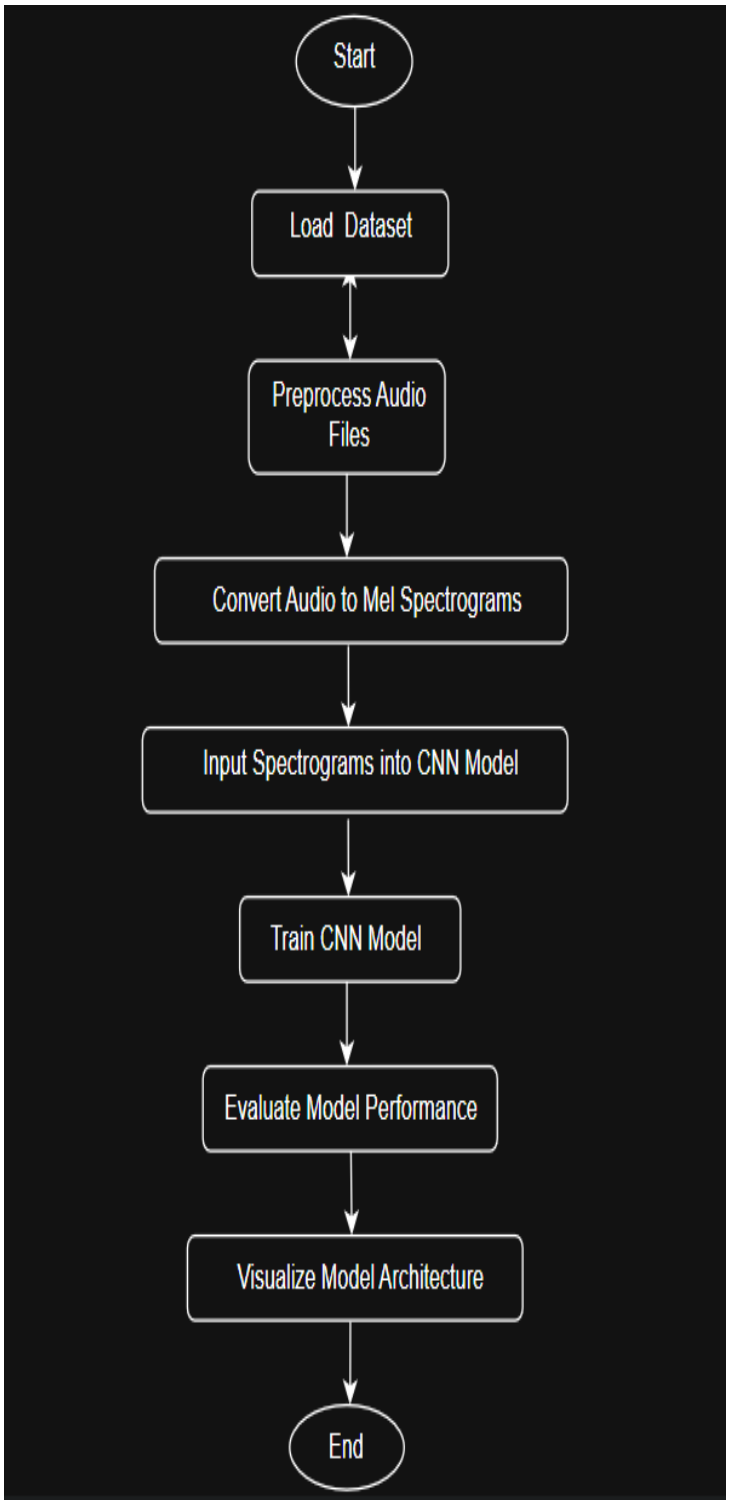

Figure 1: Work Flow of Model

Results

1.Trained Model Evaluation

The trained CNN model, loaded from deepfake_detection.h5, was evaluated on a set of real audio files. The model architecture consists of two convolutional layers, max-pooling layers, a flatten layer, a dense layer with dropout, and a softmax output layer for classifying audio as "bonafide" (real) or "spoof" (fake). The model summary is shown in Figure 2.

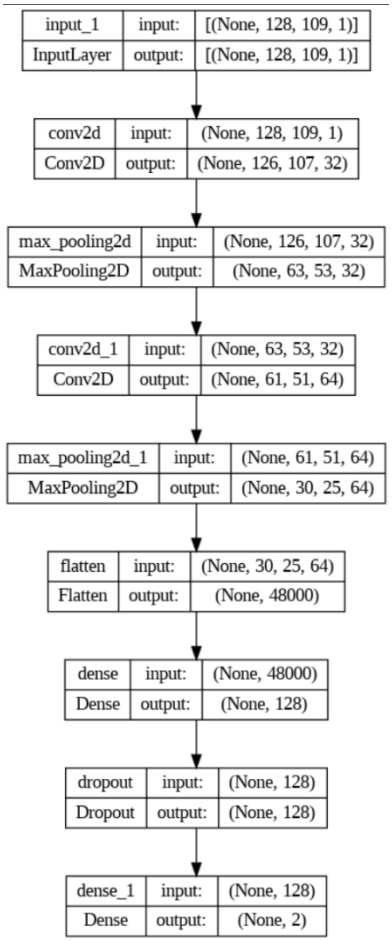

Figure 2: Model Architecture Summary

The class distribution histogram reveals an imbalanced dataset with 7 "spoof" and 3 "bonafide" instances. The confusion matrix indicates the model correctly identified 5 out of 7 "spoof" instances and all 3 "bonafide" instances, with 2 "spoof" instances misclassified as "bonafide." The precision-recall curve demonstrates an average precision of 0.87, starting at a precision of 1.0 for low recall, dropping to 0.5 at a recall of 0.6, and stabilizing near 0.6 for higher recall. The ROC curve shows strong performance with an AUC of 0.90, reflecting a high true positive rate and low false positive rate. Additionally, the calibration curve closely aligns with the ideal, indicating reliable probability estimates.

Figure 3: Class Distribution

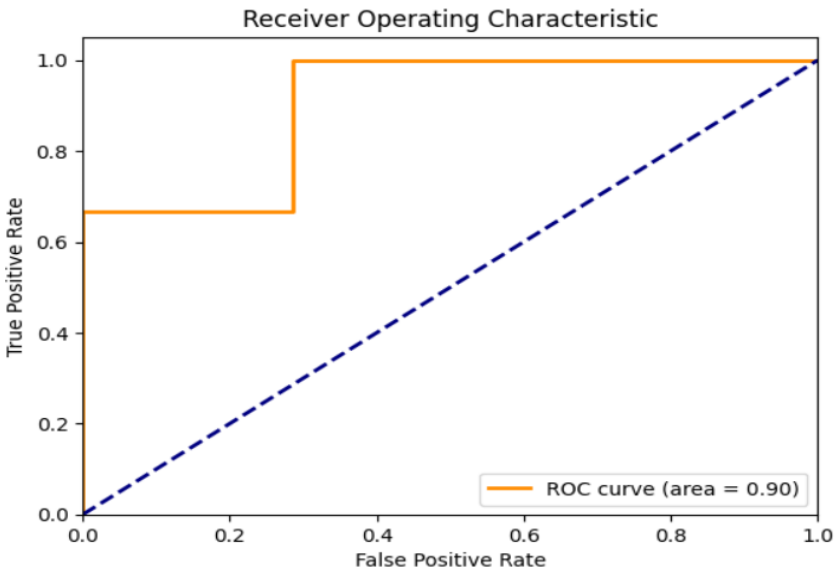

Figure 4: ROC Curve

Conclusion

The pre-trained model in test.ipynb demonstrated excellent performance in detecting real audio, correctly classifying all test samples as bonafide with high confidence. This suggests that the model is well-suited for identifying authentic audio in controlled settings. However, the inability to preprocess data at a higher level highlights the importance of verifying dataset size. Future work should focus on addressing these limitations by using more computational resources, validating the dataset structure, or employing a different dataset to enable successful model training and further evaluation.

- The model exhibits robust performance with an ROC AUC of 0.90 and an average precision of 0.87, suitable for audio spoof detection.
- Future improvements could focus on reducing false negatives for "spoof" instances to enhance overall accuracy.

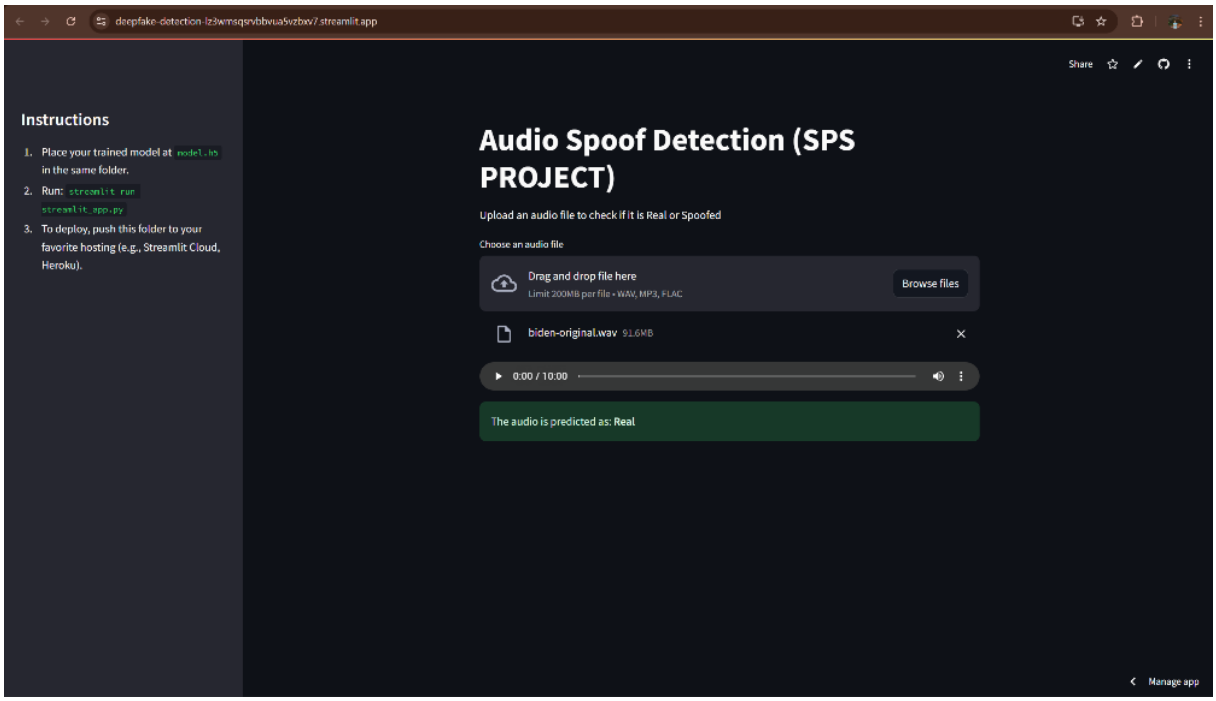

Figure 1: Real Classification

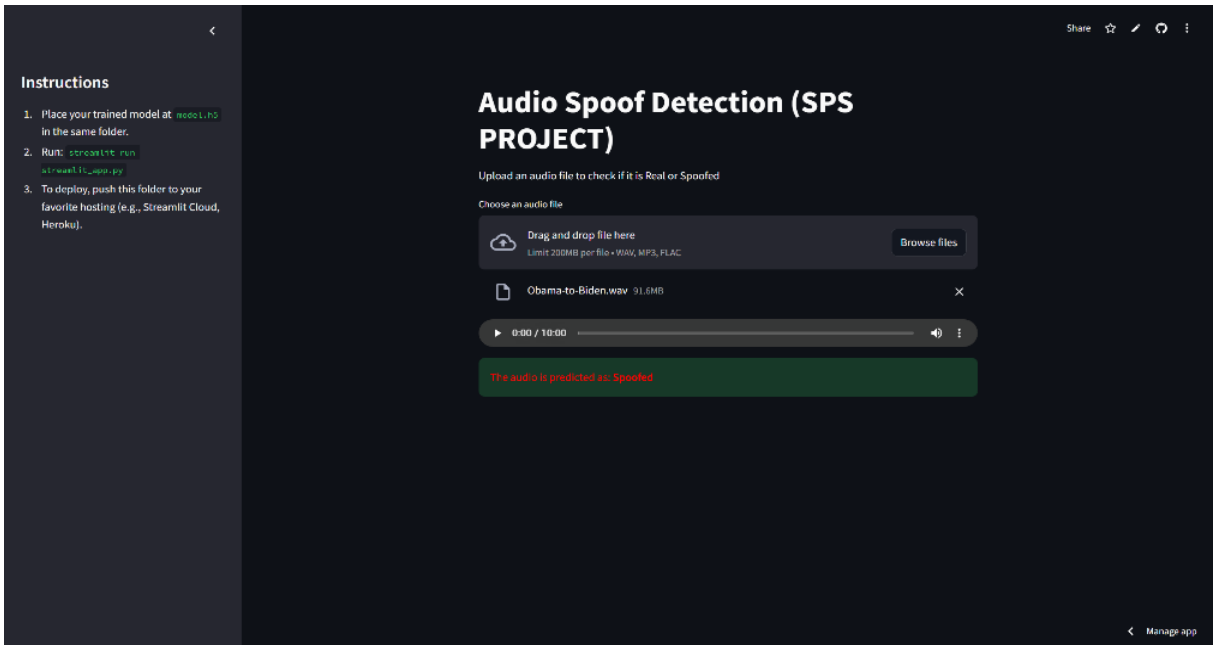

Figure 2: Spoofed Classification

References

1. Shaaban & Yildirim, "Audio Deepfake Detection Using Deep Learning," *Engineering Reports*, 2025
2. Li et al., "Audio Anti-Spoofing Detection: A Survey," *Computing Surveys*, 2024
3. Luo & Sivasundari, "Whisper+AASIST for DeepFake Audio Detection," *Proc. HCII*, 2024
4. Gao et al., "Generalized Spoofing Detection Inspired from Audio Generation Artifacts," *Interspeech*, 2021
5. Neelima & Prabha, "Hybrid Feature Optimization for Voice Spoof Detection Using CNN-LSTM," *IET Signal Processing*, 2024
6. Kashif et al., "Deepfake Audio Detection via Mel-spectrograms Using Deep CNN," *arXiv*, 2020
7. Bajwa et al., "Mel Spectrogram-Based CNN Framework for Explainable Audio Deepfake Detection," *AINA*, 2025
8. Jung et al., "AASIST: Audio Anti-Spoofing Using Integrated Spectro-Temporal Graph Attention Network," *Interspeech*, 2021
9. Martín-Doñas & Álvarez, "VICOMTECH Audio Deepfake Detection System (ADD 2022)," *ICASSP*, 2022
10. Pham et al., "Deepfake Audio Detection Using Spectrogram-based Features and Ensemble of Deep Learning Models," *arXiv*, 2024
11. Ravanelli & Bengio, "Anti-Spoofing with SincNet (ASSERT)," *Interspeech*, 2019
12. Wang & Hansen, "Toward Improving Synthetic Audio Spoofing Detection Robustness via Meta-Learning and Adversarial Training," *IEEE Access*, 2024
13. Yi et al., "ADD 2022: The First Audio Deep Synthesis Detection Challenge," *ICASSP*, 2022