PROJECT ABSTRACT

Voice Biometric Authentication using Machine Learning

Passwords are the weakest link in everyday security: they are forgotten, reused, phished and shared.

01 Title

Voice Biometric Authentication using Machine Learning

02 Introduction / Background

Passwords are the weakest link in everyday security: they are forgotten, reused, phished and shared. Voice biometrics offers a hands-free alternative — the human voice carries speaker-specific characteristics shaped by vocal-tract anatomy that are difficult to imitate. It is also an ideal final-year project: it demands the full audio-ML pipeline — signal preprocessing, feature engineering, speaker modelling and biometric evaluation — giving the student engineering to defend in the viva.

03 Problem Statement

Password fatigue pushes users toward weak, reused credentials, while commercial voice-authentication systems are proprietary black boxes. This project builds an open, examinable speaker-verification system in Python — MFCC feature extraction with librosa, per-speaker GMM models and a CNN voiceprint classifier, evaluated with standard biometric metrics (FAR, FRR, EER) and basic anti-spoofing checks.

04 Objectives

- Full speaker-verification pipeline: enrollment, MFCC extraction, model scoring, accept/reject decision.
- Per-speaker GMM models plus a CNN voiceprint-embedding classifier for comparison.
- Text-independent verification with ~10 s enrollment per user.
- Threshold selection at the equal-error-rate (EER) point.
- Replay-attack heuristics from spectral and energy cues.
- VoiceKey Streamlit app: enroll, verify, score gauge, logs, metrics.

05 Proposed System / Working

Implemented in Python 3.9+ with librosa, scikit-learn and TensorFlow. Audio is captured at 16 kHz; preprocessing applies a noise gate, voice-activity detection to drop silence, and amplitude normalization. MFCC features (40 coefficients plus delta and delta-delta, 25 ms frames with 10 ms

hop) are extracted with librosa. A GMM with 16–64 mixtures is trained per enrolled speaker; the CNN alternative maps each utterance to a 128-dimensional voiceprint embedding. Verification scores a new utterance against the claimed user's model — log-likelihood ratio for GMM, cosine similarity for the CNN — and compares the score with a threshold set at the EER point. Replay heuristics flag spectral-flatness and energy-variation anomalies typical of loudspeaker playback. The VoiceKey Streamlit app handles enrollment, live verification, attempt logging in SQLite and metric plots (DET curve, score histograms).

On the project test set the system reaches ~92–96% verification accuracy with an EER of ~4–8% depending on noise conditions and enrollment length. The live VoiceKey demo enrolls a new speaker in ~10 seconds and verifies attempts instantly, showing the similarity score gauge, accept/reject verdict and the attempt history — an impostor attempt visibly failing is the crowd-pleasing moment.

06 Technologies / Components Used

Python · librosa · scikit-learn · TensorFlow · Streamlit

07 Key Features

- Voiceprint enrollment from ~10 s of speech per user.
- 120-dim MFCC features (40 + delta + delta-delta) via librosa.
- Per-speaker GMMs (16–64 mixtures) plus CNN embedding classifier.
- Text-independent: no fixed passphrase required.
- EER-based threshold with FAR/FRR tables and DET curve.
- Replay-attack heuristics for anti-spoofing.
- VoiceKey app: enroll, verify, score gauge, logs.

08 Applications / Use Cases

- Voice-based login for apps and devices
- call-center caller verification
- smart-home personalization
- attendance systems
- hands-free authentication for accessibility
- multi-factor authentication (voice + OTP)

10 Conclusion

VoiceKey is a complete, examinable speaker-verification system in Python — from raw microphone audio to a scored accept/reject decision — with honest biometric metrics the student can defend in the viva.