ROBUST DEEPFAKE DETECTION USING MULTIMODAL ANALYSIS

ATIQUE SHAHRIER CHAKLADER

PROBLEM STATEMENT

The Problem

Deepfakes pose a growing threat to:

- Security: Misinformation, identity theft.
- Privacy: Misuse of personal images and videos.
- Public Trust: Spread of fake news.

Current detection systems:

- Focus on a single modality (visual or audio).
- Are vulnerable to advanced techniques.

Key Challenge

Need for a robust system combining multiple modalities for reliable detection.

PROJECT OBJECTIVE

Goal

Develop a Python-based deepfake detection system using multimodal analysis.

Core Features

Visual Analysis: Detect artifacts in video frames.
Audio Analysis: Identify synthetic voice and lip-sync mismatches.
Temporal Analysis: Analyze motion and frame consistency.
Multimodal Fusion: Combine all modalities for better accuracy.
User-Friendly Interface: App for video analysis and detailed reports.

PROPOSED FEATURES

Key Features

- Visual Analysis: Facial inconsistencies, blinking, lighting mismatches.
- Audio Analysis: Spectrogram analysis, voice anomalies.
- Temporal Analysis: Frame-to-frame motion analysis.
- Explainable AI: Visual heatmaps and spectrogram highlights.
- Robustness: Resilient against adversarial attacks.
- Deployment: Web/Desktop app with user-friendly interface.

DATASETS

Free and Open-Access Datasets

1.DeepFake Detection Challenge Dataset (DFDC):

High-quality real and fake videos.

2.FakeAVCeleb Dataset:

Audio-visual deepfake examples.

3.Celeb-DF:

High-quality facial deepfake videos.

4.VoxCeleb:

Audio dataset for voice analysis.

5.OpenSLR:

Speech datasets for synthetic voice detection.

TECHNOLOGIES AND TOOLS

Programming Languages

1. Python

Frameworks and Libraries

1. Deep Learning: TensorFlow, PyTorch
2. Visual Analysis: OpenCV, Dlib, Mediapipe
3. Audio Analysis: Librosa, Matplotlib
4. Multimodal Fusion: Hugging Face Transformers, Scikit-learn
5. Deployment: Flask (web app), Tkinter (desktop app)
6. Explainable AI: SHAP, Grad-CAM

Free Resources

1. Cloud Training: Google Colab (Free Tier)
2. Deployment Hosting: Heroku (Free Tier), Streamlit Cloud

IMPLEMENTATION PLAN

1. Preprocessing and Training Models

Visual Analysis:

- Train CNN (e.g., MobileNet, EfficientNet) on DFDC and Celeb-DF.

Audio Analysis:

- Use Librosa for audio feature extraction.
- Train CNN on spectrograms for voice artifact detection.

Temporal Analysis:

- Train RNN or Transformer on sequences of video frames.

Multimodal Fusion:

- Combine outputs using late fusion or attention mechanisms.

2. Deployment

- Flask API for backend processing.
- User-friendly GUI (Tkinter for desktop or Flask for web).

APP WORKFLOW

Step-by-Step Process

1. Input: User uploads a video.
2. Processing:
   - Visual, audio, and temporal models analyze the content.
   - Multimodal fusion combines results.
3. Output:
   - Display:
     Visual heatmaps (anomalies in frames).
     Spectrogram highlights (audio inconsistencies).
   - Confidence score for detection.

Deployment

- Web: Flask app hosted on Heroku or Streamlit Cloud.
- Desktop: Packaged using PyInstaller.

EXPECTED OUTCOMES

Deliverables

- Functional app for deepfake detection.
- Detailed reports with explainable insights.
- Open-source codebase for academic use.
- Evaluation report with metrics:
  Precision, Recall, F1-score, AUC.

Impact

- Strengthens trust in digital media.
- Assists journalists, law enforcement, and content moderators.

CHALLENGES AND SOLUTIONS

Challenges

1. Dataset Imbalance:
   - Real vs. fake samples.
2. Adversarial Attacks:
   - Tampered deepfakes.
3. Limited Hardware:
   - Training large models.

Solutions

1. Data Augmentation:
   - Crop, resize, add noise.
2. Adversarial Training:
   - Use modified deepfakes for robustness.
3. Cloud Resources:
   - Google Colab Free Tier.

TIMELINE

- **Project Milestones**
- Week 1-2: Dataset collection and preprocessing.
- Week 3-5: Model training (visual, audio, temporal).
- Week 6: Multimodal fusion.
- Week 7: App development (backend, GUI).
- Week 8: Testing and deployment.
- Week 9: Final report and presentation.

CONCLUSION

- Key Takeaways
- Deepfake detection requires a multimodal approach for robustness.
- This project combines free resources to develop a comprehensive solution.
- Expected impact: Improved trust in digital media and enhanced tools for detection.