

Deepfake Detection Using CNN and RNN

1. Introduction

This project focuses on building a deepfake detection system using CNN (for video analysis) and RNN (for audio analysis). We utilized the Celeb-DF-v2 dataset for videos and Deepfake Audio datasets for audio detection. Frameworks like TensorFlow and PyTorch were used along with OpenCV, Librosa, and FFmpeg.

2. Dataset Information

- Celeb-DF-v2 Dataset: Real and deepfake videos.

Link: <https://arxiv.org/abs/2003.07590>

- Deepfake Audio Dataset: Real and AI-generated voices.

Example source: https://github.com/AsaduzzamanCSE/deepfake_audio_dataset

3. Methodology

- Video-based Detection:
 - Extract frames from videos.
 - Resize frames to 224x224.
 - Train CNN (MobileNetV2 / ResNet) on frame images.
- Audio-based Detection:
 - Extract audio from videos using FFmpeg.
 - Convert audio into Mel spectrograms using Librosa.
 - Train RNN (LSTM) on spectrogram features.

4. Tools and Libraries Used

- TensorFlow and Keras: <https://www.tensorflow.org/>
- PyTorch: <https://pytorch.org/>
- OpenCV: <https://opencv.org/>
- Librosa: <https://librosa.org/>

Deepfake Detection Using CNN and RNN

- FFmpeg: <https://ffmpeg.org/>

5. Model Architectures

- CNN Model:
Based on MobileNetV2 / ResNet50 architectures.
- RNN Model:
Bi-directional LSTM layers trained on 128 Mel-spectrogram inputs.

6. Training Details

- Video Dataset:
 - Frames extracted and balanced.
 - Trained CNN using class weights.
- Audio Dataset:
 - Generated spectrograms from audio clips.
 - Trained RNN with appropriate hyperparameters.

7. Prediction System

The final system evaluates both video frames and audio separately, predicting whether the sample is Real or Fake based on aggregation of frame and audio predictions. Outputs are saved in a CSV file for report generation.

8. Literature and References

- Celeb-DF-v2 Dataset Paper: <https://arxiv.org/abs/2003.07590>
- MobileNetV2 Paper: <https://arxiv.org/abs/1801.04381>
- ResNet Paper: <https://arxiv.org/abs/1512.03385>
- TensorFlow: <https://www.tensorflow.org/>
- PyTorch: <https://pytorch.org/>

Deepfake Detection Using CNN and RNN

- OpenCV: <https://opencv.org/>
- Librosa: <https://librosa.org/>
- FFmpeg: <https://ffmpeg.org/>