International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 11 | Nov - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Real-Time Deepfake Detection in Online Communication: A Review of AI-Based Audio-Visual Detection Techniques and a Proposed Yii2–MySQL Framework

Priyabrata Nayak[1]

[1] Department of Computer Science and Engineering, Nalanda Institute of Technology, Bhubaneswar, Odisha, India.

nayak.priyabrata83@gmail.com

Abstract

The rapid advancement of artificial intelligence (AI) and generative models has increased the sophistication of manipulated audio and video content, creating significant challenges for the authenticity of online communication. Real-time deepfake detection has therefore become an important research area in multimedia security and digital forensics. This paper presents a literature-based review of AI techniques for detecting video deepfakes, synthetic speech and multimodal audio-visual manipulation. It examines convolutional neural networks, transformer-based architectures, acoustic feature analysis, temporal inconsistency detection and cross-modal feature fusion. Particular attention is given to detection latency, computational requirements, media compression, robustness and generalization to previously unseen manipulation techniques. Based on the reviewed literature, the paper proposes a conceptual web-based integration framework using the Yii2 PHP framework and MySQL database, supported by an independent machine-learning inference service. The proposed architecture separates multimedia analysis from application management to support detection requests, result storage and monitoring. The study identifies research gaps and discusses future directions for reliable, efficient and privacy-aware deepfake detection in online communication environments.

Keywords: Deepfake Detection, Audio Deepfake, Video Deepfake, Multimodal AI, Real-Time Communication, Yii2 Framework.

I. INTRODUCTION

1.1 Background

Artificial Intelligence (AI), particularly deep learning and generative models, has transformed the creation and distribution of digital audio and video content.

Deepfakes are audio, video or image content generated or manipulated using AI techniques to imitate individuals or alter existing media.

The increasing realism of deepfakes raises concerns about misinformation, identity impersonation, digital fraud and privacy.

1.2 Video Deepfake Detection

Video deepfake detection aims to identify artificially generated or manipulated visual content.

Transformer-based architectures and frequency-domain analysis have also been investigated to capture relationships between image regions and identify manipulation-related characteristics.

1.3 Audio Deepfake Detection

Audio deepfake detection focuses on identifying artificially generated or manipulated speech.

Detection methods examine acoustic representations such as Mel-Frequency Cepstral Coefficients (MFCCs), Linear Frequency Cepstral Coefficients (LFCCs), Constant-Q Cepstral Coefficients (CQCCs) and spectrograms.

Datasets such as ASVspoof have supported research into speech spoofing detection.

1.4 Need for Multimodal Deepfake Detection

Video and audio detection provide complementary evidence for assessing multimedia authenticity.

Multimodal audio-visual detection combines visual and acoustic information and can examine relationships between facial movements and corresponding speech.

1.5 Problem Statement

Although existing research has developed several methods for detecting manipulated videos and synthetic audio, important challenges remain in applying these techniques to real-time online communication.

Computational requirements and processing latency can also restrict the use of complex detection models in interactive environments.

This study reviews AI-based video, audio and multimodal deepfake detection techniques and proposes a conceptual web application framework using the Yii2 PHP framework and MySQL database.

1.6 Research Objectives

The specific objectives are:

1. To review AI-based techniques for video deepfake detection.
2. To examine synthetic speech and audio deepfake detection methods.
3. To analyse multimodal audio-visual detection and synchronization-based approaches.

4. To investigate real-time detection challenges, including latency, computational requirements and model generalization.
5. To compare detection techniques, benchmark datasets and evaluation metrics reported in existing literature.
6. To propose a conceptual Yii2–MySQL application architecture for integrating independent AI inference services and identify relevant research gaps.

1.7 Scope and Contributions

This study reviews AI-based detection of manipulated videos, synthetic audio and combined audio-visual deepfakes.

The proposed framework separates application management from multimedia inference.

The major contributions of this study are as follows:

1. A review of AI-based video and audio deepfake detection techniques.
2. An analysis of multimodal audio-visual detection and synchronization approaches.
3. A comparison of detection methods, datasets, evaluation metrics and real-time processing challenges.
4. A conceptual Yii2–MySQL architecture for integrating external AI-based detection services.
5. Identification of research gaps and future research opportunities.

The proposed application framework is conceptual and is presented as an architectural design.

1.8 Organization of the Paper

The remainder of this paper is organized as follows.

II. REVIEW METHODOLOGY

2.1 Review Design

This study adopts a structured literature review approach to examine AI-based techniques for detecting video deepfakes, synthetic audio and multimodal audio-visual manipulation.

It also examines the conceptual integration of existing detection techniques into a web-based application using the Yii2 PHP framework and MySQL database.

2.2 Literature Sources and Search Strategy

The literature search was conducted using IEEE Xplore and Google Scholar to identify research on AI-based video deepfake detection, audio deepfake detection, multimodal audio-visual analysis and real-time detection.

The review primarily covers research published between 2020 and 2026.

2.3 Search Terms and Search Expressions

The search strategy used combinations of keywords covering deepfake generation, AI-based detection, individual detection modalities, multimodal analysis and real-time processing.

Representative Boolean search expressions include:

Search Expression 1: Video Detection

("deepfake detection" OR "video manipulation detection") AND ("deep learning" OR CNN OR transformer)

Search Expression 2: Audio Detection

("audio deepfake" OR "synthetic speech detection" OR "voice cloning detection") AND ("machine learning" OR "deep learning")

Search Expression 3: Multimodal Detection

("audio-visual deepfake detection" OR "multimodal deepfake detection") AND ("feature fusion" OR synchronization OR transformer)

Search Expression 4: Real-Time Detection

("real-time deepfake detection" OR "online deepfake detection") AND (latency OR efficiency OR streaming)

Search Expression 5: Application Integration

("deepfake detection framework" OR "multimedia detection system") AND (web application OR API OR deployment)

The search expressions were applied using the search facilities of IEEE Xplore and Google Scholar.

2.4 Inclusion Criteria

Publications were considered for inclusion when they addressed one or more of the following areas:

1. AI or machine-learning techniques for detecting manipulated video, synthetic audio or audio-visual deepfakes.
2. Relevant detection architectures, feature extraction methods, classification techniques or multimodal fusion strategies.
3. Evaluation using recognized datasets or clearly described experimental data.
4. Real-time processing, computational efficiency, robustness, generalization or deployment challenges.
5. Multimedia security, digital forensics or integration of AI-based detection services.

Peer-reviewed journal articles and conference papers were prioritized.

2.5 Exclusion Criteria

Publications were excluded from detailed consideration when they:

1. Did not address deepfake generation, manipulation detection or a directly related multimedia security problem.
2. Focused exclusively on synthetic content generation without relevant information about detection.
3. Lacked sufficient methodological or technical information for meaningful analysis.
4. Were duplicate versions of another research work.

5. Fell outside the audio, video and multimodal detection scope.
6. Were unverified webpages, promotional articles or non-academic discussions without sufficient technical evidence.

Where multiple versions of a research work were available, the most complete and reliable version was preferred.

2.6 Literature Screening and Selection

The literature was screened qualitatively for relevance to the research objectives and the technical scope of this review.

2.7 Data Extraction and Classification

Information from the selected publications was organized according to publication year, research objective, detection modality, model architecture, feature extraction method, dataset, evaluation metrics and major findings.

The reviewed studies were classified into four categories:

1. **Video deepfake detection:** CNN-based methods, temporal analysis, transformer architectures, frequency-domain approaches and hybrid techniques.
2. **Audio deepfake detection:** Acoustic feature analysis, spectrogram-based methods, raw-waveform processing and deep-learning architectures.
3. **Multimodal audio-visual detection:** Early fusion, late fusion, hybrid fusion, attention mechanisms and audio-visual synchronization.
4. **Real-time detection and deployment:** Inference latency, computational complexity, media compression, streaming, scalability and application integration.

This classification supports the qualitative comparison of detection approaches and their practical applicability.

2.8 Comparative Analysis Approach

The selected publications were compared according to their detection objectives, technical approaches, experimental settings and reported evaluation measures.

For multimodal detection, the comparison considers individual modality contributions, fusion strategies, audio-visual synchronization and the effects of missing or degraded input streams.

Performance values from different studies are not treated as directly comparable unless their datasets, experimental protocols and evaluation conditions are sufficiently consistent.

2.9 Methodological Limitations

This review has several limitations.

The proposed Yii2–MySQL framework is a conceptual application architecture derived from requirements identified in the reviewed literature.

III. BACKGROUND AND RELATED WORK

3.1 Evolution of Deepfake Technology

Deepfake technology has evolved through advances in machine learning, deep neural networks and generative artificial intelligence.

Generative Adversarial Networks (GANs) contributed significantly to synthetic media generation through the interaction of a generator and a discriminator [1].

In audio generation, neural Text-to-Speech systems, voice-conversion models and neural vocoders have improved the naturalness of synthetic speech.

3.2 Major Categories of Deepfake Manipulation

Deepfake manipulation can be classified according to the type of multimedia content being modified and the underlying generation technique.

A. Video and Facial Manipulation

Video manipulation includes face swapping, facial reenactment, facial attribute modification, synthetic face generation and lip-synchronization manipulation.

B. Audio Manipulation

Audio manipulation includes Text-to-Speech (TTS) synthesis, voice cloning, voice conversion, modification of recorded speech and synthetic audio generation.

C. Multimodal Audio-Visual Manipulation

Multimodal deepfakes manipulate more than one media modality.

3.3 AI Technologies Used in Deepfake Detection

AI-based deepfake detection uses neural architectures and signal-processing techniques to identify patterns associated with manipulated multimedia.

A. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) extract spatial features from images, video frames and audio spectrograms.

B. Recurrent Neural Networks

Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures, process sequential information.

C. Transformer-Based Architectures

Transformer-based architectures use attention mechanisms to capture relationships between different parts of an input sequence.

D. Frequency-Domain Analysis

Frequency-domain methods examine signal characteristics that may not be readily visible in ordinary image or waveform representations.

E. Multimodal Feature Fusion

Multimodal detection combines information from audio and video.

The suitability of each approach depends on the detection task, dataset, manipulation method and computational resources.

3.4 Existing Research and Related Studies

Several studies have contributed to the development of deepfake detection methods and benchmark evaluation.

Rössler et al. introduced FaceForensics++, a dataset containing authentic videos and facial manipulations produced using methods such as DeepFakes, Face2Face, FaceSwap and NeuralTextures. Different compression levels support the evaluation of detection methods under varying video-quality conditions [2].

Li et al. introduced Celeb-DF to provide more realistic manipulated face videos for deepfake detection research [3]. Jiang et al. proposed DeeperForensics-1.0, which includes variations in subjects, facial expressions, poses and simulated distortions to support research into real-world face forgery detection [4].

In speech anti-spoofing research, ASVspoof provides benchmark data for evaluating countermeasures against synthetic, converted and replayed speech [5]. AASIST applies integrated spectro-temporal graph attention networks to audio anti-spoofing, demonstrating the use of relationships between spectral and temporal information [6].

Multimodal research has investigated the combination of audio and visual evidence. FakeAVCeleb provides manipulated audio-visual content for studying facial and speech manipulation [7], while AV-Deepfake1M extends research towards larger and more diverse audio-visual manipulation scenarios [8].

These studies provide important foundations for video, audio and multimodal deepfake detection.

3.5 Benchmark Datasets for Deepfake Detection

Benchmark datasets are essential for training, evaluating and comparing deepfake detection methods.

The following datasets are relevant to this study:

A. FaceForensics++

FaceForensics++ contains authentic videos and manipulated versions produced using multiple facial manipulation methods. It supports visual artefact detection research and evaluation under different compression conditions [2].

B. Celeb-DF

Celeb-DF provides realistic manipulated face videos and is used to evaluate detection methods against high-quality face-swapping manipulations [3].

C. DeepFake Detection Challenge Dataset

The DeepFake Detection Challenge (DFDC) dataset contains authentic and manipulated videos for evaluating video deepfake detection methods under varied recording and manipulation conditions [17].

D. DeeperForensics-1.0

DeeperForensics-1.0 includes variations in subjects, facial expressions, poses and video distortions. It is relevant to robustness and generalization studies [4].

E. ASVspoof

ASVspoof datasets support research into automatic speaker verification security and speech anti-spoofing. ASVspoof 2019 includes logical-access and physical-access scenarios involving synthetic, converted and replayed speech [5].

F. WaveFake

WaveFake supports synthetic speech detection research across speech generated using different neural synthesis approaches [18].

G. FakeAVCeleb

FakeAVCeleb contains manipulated audio-visual content and is relevant to research involving synthetic speech and manipulated facial content [7].

H. AV-Deepfake1M

AV-Deepfake1M provides a large-scale audio-visual deepfake benchmark for studying multimodal manipulation detection [8].

The datasets differ in media types, manipulation techniques, sample characteristics and evaluation protocols.

3.6 Limitations of Existing Research

Despite progress in deepfake detection, several limitations remain.

1. Generalization: Models trained on specific benchmark datasets may not perform reliably against previously unseen manipulation techniques.
2. Media quality: Video compression, resizing and filtering can affect visual artefacts, while background noise, microphone characteristics and audio codecs can influence acoustic features.
3. Computational requirements: Analysing multiple video frames and audio segments can increase inference time and memory consumption, restricting continuous processing.
4. Unimodal evaluation: Video and audio detection are often evaluated independently, limiting the analysis of inconsistencies between visible facial movements and associated speech.
5. **Multimodal complexity:** Synchronization, missing input streams, unequal modality quality and feature-fusion costs create additional challenges.
6. Application-level deployment: Research often emphasizes model development and benchmark evaluation rather than complete application integration.

3.7 Research Gap and Motivation for the Proposed Framework

The reviewed literature demonstrates considerable progress in video, audio and multimodal deepfake detection.

A practical application must coordinate media acquisition, preprocessing, AI inference, multimodal result integration and response delivery while managing processing delays.

This study therefore proposes a conceptual framework using Yii2 for backend application management and MySQL for storing user information, detection requests, processing metadata and results.

3.8 Chapter Summary

This section reviewed the evolution of deepfake technology, major categories of multimedia manipulation, AI technologies used in detection and relevant benchmark datasets.

IV. AI-BASED VIDEO DEEPFAKE DETECTION

4.1 Overview of Video Deepfake Detection

Video deepfake detection identifies manipulated or artificially generated visual content by analysing spatial, temporal and statistical characteristics of video sequences.

AI-based approaches learn patterns associated with facial inconsistencies, image artefacts and temporal irregularities.

4.2 Video Frame Extraction and Preprocessing

Video preprocessing prepares incoming frames for analysis and can influence detection performance.

A typical preprocessing pipeline includes:

1. Video decoding and frame extraction.
2. Frame sampling at selected intervals.
3. Face detection and facial-region localization.
4. Facial-region cropping and resizing.
5. Pixel normalization and image preparation.
6. Organization of frames into individual samples or temporal sequences.

Preprocessing should preserve manipulation-related information.

4.3 CNN-Based Video Deepfake Detection

Convolutional Neural Networks (CNNs) extract spatial features from images and video frames.

Several CNN architectures have been investigated for facial manipulation detection:

A. XceptionNet

XceptionNet uses depthwise separable convolutions to learn visual representations with a comparatively efficient convolutional structure. It has been widely used as a baseline architecture in facial manipulation detection research, including work involving the FaceForensics++ dataset [9].

B. ResNet

Residual Networks use skip connections to support the training of deeper neural networks. ResNet-based architectures can extract hierarchical visual features and have been investigated for image and video manipulation classification [10].

C. EfficientNet

EfficientNet uses compound scaling to balance network depth, width and input resolution, while Inception-based networks use parallel convolutional operations to capture patterns at different spatial scales.

CNN-based methods provide a foundation for frame-level detection.

4.4 Temporal and Sequence-Based Detection

Video contains temporal information that cannot be fully represented by isolated frames.

A. Recurrent Neural Networks

Recurrent Neural Networks (RNNs) process sequential visual features and can model changes across consecutive video frames.

B. Long Short-Term Memory Networks

Long Short-Term Memory (LSTM) networks use memory cells and gating mechanisms to model temporal dependencies.

C. Gated Recurrent Units

Gated Recurrent Units (GRUs) provide another approach to modelling sequential information, with a comparatively simpler gating structure than LSTMs.

D. Three-Dimensional Convolutional Networks

Three-dimensional CNNs apply convolution operations across spatial and temporal dimensions, allowing short video sequences to be analysed jointly.

Temporal approaches complement frame-based CNN models, but their computational requirements increase with sequence length, frame resolution and model complexity.

4.5 Transformer-Based Video Deepfake Detection

Transformer-based architectures use attention mechanisms to model relationships between different parts of an input [11]. Vision Transformers divide images into patches and represent them as tokens, allowing the model to learn relationships between image regions [12].

For video analysis, transformer-based models can also examine temporal relationships across frame sequences.

4.6 Frequency-Domain and Artifact-Based Detection

Frequency-domain analysis examines image characteristics that may not be readily visible in ordinary spatial representations.

Artifact-based detection may examine facial blending boundaries, skin texture, illumination, facial geometry, expression transitions and unusual frequency characteristics.

4.7 Video Deepfake Detection in Real-Time Communication

Real-time video deepfake detection must analyse incoming frames while maintaining an acceptable response time.

The main deployment considerations include:

A. Frame Sampling: Analysing selected frames can reduce computational cost, but excessive sampling reduction may miss short-lived manipulation events.

B. Lightweight Models: Compact architectures, quantization, pruning and knowledge distillation may reduce inference requirements. Their effect on detection reliability requires evaluation.

C. Temporal Window Management: Sequence-based models require an appropriate number of frames to balance temporal information and processing delay.

D. Hardware Acceleration: GPUs and other supported acceleration technologies may improve inference speed, depending on the model and deployment environment.

E. Network and Compression Effects: Online platforms may compress, resize or otherwise transform video streams, affecting the visual features used by detection models.

F. Continuous Processing: A real-time system must repeatedly analyse incoming video segments and manage processing queues, inference requests and result delivery.

Real-time suitability therefore depends on the complete processing pipeline rather than classification performance alone.

4.8 Proposed Video Detection Workflow for the Integrated Framework

This study proposes a conceptual video detection workflow that separates web application management from AI-based inference.

The proposed workflow consists of:

1. Video Acquisition: Receiving an authorized video input or controlled reference to a communication stream.

2. Frame Extraction: Selecting frames according to a configured sampling strategy.

3. Preprocessing: Decoding, resizing and normalizing the selected frames.

4. Feature Extraction: Extracting spatial, temporal or frequency-domain features.

5. Classification: Producing a prediction using a video deepfake detection model.

6. Result Generation: Preparing the classification output, processing time and confidence information where available.

7. Application Integration: Transmitting the result to the Yii2 backend through an authenticated API.

8. Result Storage: Recording relevant request information and detection metadata in MySQL.

The conceptual processing flow is:

Video Input → Frame Extraction → Preprocessing → AI-Based Video Detection → Result Generation → Yii2 Backend → MySQL Database and User Dashboard.

This workflow is a proposed architectural design and does not represent an implemented or experimentally validated detection pipeline.

4.9 Comparative Analysis of Video Detection Approaches

Video deepfake detection approaches differ in the visual information they analyse, their ability to identify temporal inconsistencies and their computational requirements.

TABLE 4.1: COMPARATIVE ANALYSIS OF VIDEO DEEPFAKE DETECTION APPROACHES

Method	Features	Strength	Limit	Real-time use
CNN (Xception, ResNet)	Face texture; blends; spatial cues	Good frame features	Weak time context; compression-sensitive	Good with light models
EfficientNet	Scaled spatial features	Flexible model size	Needs speed and domain checks	Possible after tuning
RNN / LSTM / GRU	Face and motion over time	Models sequences	More frames add delay	Use short windows
3D CNN	Space and time in clips	Joint clip analysis	Higher compute and memory	Hardware-dependent
Vision Transformer	Patch and frame links	Broad context	High compute and memory	Needs optimization
FFT / DCT	Image frequency patterns	Adds spectral cues	Affected by compression	Use with other cues
Artifact analysis	Edges; light; skin; shape	Finds visible flaws	Flaws may be absent	Use as support

Source: Comparative synthesis of the video detection approaches discussed in this section, including the architectures and techniques reported in [9]–[12].

The comparison indicates that CNN-based approaches provide a foundation for spatial feature extraction, while temporal and three-dimensional CNN methods extend analysis to changes across video frames. Transformer-based architectures offer broader contextual modelling capabilities, although their computational requirements may restrict their use in resource-constrained environments [11], [12].

Frequency-domain and artifact-based techniques provide complementary information that may assist in identifying manipulation-related patterns.

For real-time communication, the selection of a detection approach must consider processing latency, memory requirements, frame sampling, video resolution and hardware availability.

4.10 Discussion

The reviewed approaches demonstrate the importance of spatial, temporal and frequency-related information in video deepfake detection.

Frequency-domain and artifact-based methods provide complementary information, but no single architecture can be assumed to perform reliably across all manipulation techniques and communication environments.

For real-time applications, model selection must consider detection capability, inference latency, memory requirements, frame sampling, video quality and hardware availability.

4.11 Section Summary

This section reviewed AI-based video deepfake detection techniques, including frame preprocessing, CNN-based architectures, temporal analysis, transformer-based approaches and frequency-domain methods.

V. AI-BASED AUDIO DEEPFAKE DETECTION

5.1 Fundamentals of Audio Deepfake Generation

Audio deepfakes are artificially generated or manipulated audio signals produced using AI techniques to imitate natural speech or reproduce the voice characteristics of an individual.

Major forms of audio manipulation include:

A. Text-to-Speech Synthesis: Text-to-Speech (TTS) systems convert written text into spoken audio. Neural TTS models can generate natural-sounding speech and may be misused to imitate individuals.

B. Voice Cloning: Voice cloning attempts to reproduce the acoustic characteristics of a particular speaker using reference speech.

C. Voice Conversion: Voice conversion modifies the perceived speaker characteristics of an existing speech signal while attempting to preserve its linguistic content.

D. Speech Manipulation: This involves modifying selected portions of an audio recording, such as replacing spoken words or altering specific acoustic segments.

E. Neural Audio Generation: Generative neural architectures and neural vocoders can produce realistic speech waveforms from learned representations.

These developments make conventional audio authenticity verification increasingly challenging.

5.2 Audio Acquisition and Preprocessing

Audio preprocessing prepares incoming recordings or communication streams for analysis.

A typical preprocessing pipeline includes:

1. Audio acquisition and decoding.
2. Sampling-rate adjustment according to model requirements.
3. Channel handling for mono or multi-channel audio.
4. Segmentation into manageable audio intervals.
5. Signal normalization where appropriate.
6. Noise handling when required by the detection model.

For real-time applications, segmentation is particularly important because continuous audio must be analysed in smaller processing windows.

5.3 Acoustic Feature Extraction

Acoustic feature extraction converts raw audio into representations suitable for machine-learning and deep-learning models.

A. Mel-Frequency Cepstral Coefficients (MFCCs): MFCCs represent short-term spectral characteristics using a perceptually motivated frequency scale. They are widely used in speech processing and audio spoofing detection.

B. Linear Frequency Cepstral Coefficients (LFCCs): LFCCs provide a cepstral representation based on a linear frequency scale and have been investigated in speech anti-spoofing research.

C. Constant-Q Cepstral Coefficients (CQCCs): CQCCs are derived from Constant-Q Transform representations and provide an alternative frequency-based representation for analysing acoustic characteristics.

D. Spectrogram and Short-Time Fourier Transform (STFT): STFT represents changes in the frequency content of audio over time. Spectrograms provide time-frequency representations that can be analysed using CNN-based models.

E. Raw Waveform Representation: Some deep-learning architectures process raw audio waveforms directly, allowing models to learn task-specific representations without relying entirely on handcrafted features.

The selection of acoustic features depends on the detection objective, dataset characteristics, model architecture and computational resources.

5.4 Deep-Learning Architectures for Audio Deepfake Detection

Deep-learning architectures identify patterns from acoustic representations or raw audio signals and use them to classify recordings.

A. CNN-Based Audio Detection: CNNs can analyse spectrograms and other time-frequency representations. They extract local spectral patterns and learn more complex acoustic characteristics. Their effectiveness depends on the input representation and the manipulation methods represented in the training data.

B. Recurrent Neural Networks: RNNs, including LSTM and GRU architectures, model temporal dependencies in speech signals and extracted acoustic features. They can analyse variations across consecutive audio frames or segments, although longer sequences may increase processing requirements.

C. RawNet2: RawNet2 is an architecture investigated in audio spoofing countermeasure research. It processes raw waveform information and learns representations directly from speech signals [13].

D. AASIST: Audio Anti-Spoofing using Integrated Spectro-Temporal Graph Attention Networks (AASIST) uses graph attention mechanisms to model relationships between spectral and temporal information for spoofed-speech detection.

E. Transformer-Based Audio Detection: Transformer-based approaches use attention mechanisms to model relationships across speech representations. Pretrained models such as wav2vec 2.0 and WavLM have been investigated as feature extractors or components in downstream speech-processing tasks [14], [15]. Their application to deepfake detection requires appropriate training and evaluation. Computational requirements must also be considered for real-time deployment.

5.5 Benchmark Datasets for Audio Deepfake Detection

Benchmark datasets provide structured samples and evaluation conditions for research on synthetic speech and audio spoofing detection.

A. ASVspoof 2019: This benchmark includes logical-access and physical-access scenarios involving synthetic, converted and replayed speech. It is particularly relevant to automatic speaker verification spoofing countermeasures, rather than every form of audio deepfake detection [5].

B. ASVspoof 2021: This dataset extends speech spoofing evaluation through additional conditions designed to investigate detection under variations in recording and transmission environments [19].

C. WaveFake: WaveFake supports synthetic speech detection research using generated speech samples from different neural speech-generation methods. It is relevant to examining generalization across synthesis techniques [18].

D. Fake-or-Real (FoR): The FoR dataset provides genuine and synthetic speech examples for investigating machine-learning-based audio authenticity classification [20].

E. In-the-Wild Audio Datasets: In-the-wild collections can help evaluate detection under varied recording conditions, background noise, transmission channels and speech-generation methods.

F. FakeAVCeleb: FakeAVCeleb contains manipulated audio-visual content and supports research involving synthetic speech alongside manipulated facial content [7].

These datasets differ in recording conditions, generation techniques, speakers and evaluation protocols.

5.6 Evaluation Metrics for Audio Deepfake Detection

Audio detection systems require evaluation measures that describe classification performance and processing suitability.

A. Accuracy: Measures the proportion of correctly classified samples among all evaluated samples.

B. Precision: Measures the proportion of samples predicted as manipulated that are actually manipulated.

C. Recall: Measures the proportion of actual manipulated samples correctly identified by the system.

D. F1-Score: Represents the harmonic mean of precision and recall.

E. Equal Error Rate (EER): Represents the operating point at which the false acceptance rate and false rejection rate are equal. It is commonly used in biometric verification and speech anti-spoofing evaluation.

F. Tandem Detection Cost Function (t-DCF): Used particularly when a speech anti-spoofing countermeasure operates together with an automatic speaker verification system. It considers the combined cost of errors from both components and is not a universal metric for all audio deepfake applications.

G. Processing Latency: Measures the time required to analyse an audio sample or segment and produce a detection result. For real-time communication, both inference time and total system response time are relevant.

The selected metrics should reflect the detection objective.

5.7 Real-Time Audio Deepfake Detection

Real-time audio deepfake detection requires incoming speech to be analysed while maintaining an acceptable processing delay.

A. Audio Segmentation: Continuous streams can be divided into short overlapping or non-overlapping segments. Smaller segments may provide faster initial results but insufficient acoustic context.

B. Streaming Feature Extraction: Acoustic features can be extracted incrementally rather than waiting for the complete recording. The process must preserve relevant temporal information.

C. Model Optimization: Lightweight architectures, quantization and other optimization techniques may reduce inference time. Their effects on detection reliability require evaluation.

D. Background Noise and Recording Conditions: Noise, microphone variation, echo, packet loss and audio compression can alter acoustic characteristics and affect detection.

E. Processing Delay: Segment duration, feature extraction, inference and result delivery must be balanced. Excessive buffering increases delay, while very short segments may reduce classification reliability.

F. Continuous Monitoring: An online application may require repeated analysis of incoming audio segments. It should associate detection results with the relevant audio intervals and manage processing requests without interrupting communication.

Real-time audio detection therefore depends on the complete processing pipeline, not only the classification model.

5.8 Proposed Audio Detection Workflow

This study proposes a conceptual audio detection workflow integrated with the Yii2–MySQL application framework.

The proposed stages are:

1. Audio Acquisition: Receive an authorized audio recording or controlled audio-stream reference.
2. Audio Decoding: Convert the input into a suitable digital audio representation.
3. Segmentation: Divide the audio stream into manageable segments.
4. Preprocessing: Apply sampling-rate adjustment, channel handling and other required preparation.
5. Feature Extraction: Generate MFCC, LFCC, CQCC, spectrogram or learned waveform representations.

6. AI-Based Detection: Process the extracted information using a trained audio detection model.
7. Classification: Produce a detection output for the analysed segment.
8. Result Generation: Prepare the classification output, model information, processing time and confidence information where available.
9. Application Integration: Transmit the result to the Yii2 backend through an authenticated API.
10. Result Storage: Record relevant request information and detection metadata in MySQL.

The conceptual processing flow is:

Audio Input → Decoding and Segmentation → Preprocessing → Acoustic Feature Extraction → AI-Based Audio Detection → Result Generation → Yii2 Backend → MySQL Database and User Dashboard.

The audio detection service is proposed as an independent machine-learning component.

5.9 Comparative Analysis of Audio Detection Approaches

Audio detection approaches differ in their input representations, learning mechanisms and computational requirements.

TABLE 5.1: COMPARATIVE ANALYSIS OF AUDIO DEEPFAKE DETECTION APPROACHES

Method	Input / feature	Strength	Limit
MFCC	Mel cepstral features	Compact speech cues	May miss advanced synthesis
LFCC	Linear cepstral features	Useful in spoof tests	Recording-sensitive
CQCC	Constant-Q features	Alternative spectral view	Task-dependent
CNN	Spectrograms	Learns local patterns	Data-dependent
RNN / LSTM	Acoustic sequences	Models speech over time	Longer processing
RawNet2	Raw waveform	Learns from audio [13]	Needs data and compute
AASIST	Spectro-temporal graph	Links spectral and time cues	More complex
Transformer	Attention features	Captures broad context	Higher compute [14], [15]

Source: Comparative synthesis of the audio detection methods and architectures discussed in this section.

The comparison summarizes general methodological characteristics rather than establishing a universal performance ranking.

5.10 Discussion

The reviewed approaches demonstrate that synthetic speech detection can use different acoustic representations and deep-learning architectures.

AASIST and transformer-based methods provide additional approaches for modelling complex acoustic relationships.

Real-time applications introduce further constraints involving segment duration, processing delay, computational resources and continuous-stream management.

5.11 Section Summary

This section examined audio deepfake generation, including text-to-speech synthesis, voice cloning, voice conversion and speech manipulation.

Relevant benchmark datasets, including ASVspoof, WaveFake, FoR and FakeAVCeleb, were reviewed alongside evaluation measures such as accuracy, precision, recall, F1-score, EER and t-DCF.

The following section examines multimodal audio-visual deepfake detection, focusing on the combination of visual and acoustic evidence.

VI. MULTIMODAL AUDIO-VISUAL DEEPFAKE DETECTION

6.1 Fundamentals of Multimodal Deepfake Detection

Multimodal audio-visual deepfake detection identifies manipulated multimedia by analysing both visual and acoustic information.

Visual information includes facial appearance, expressions, lip movements and head movements, while audio information includes speech patterns, voice characteristics and acoustic features.

A video may contain an artificially replaced face while retaining original audio.

6.2 Audio-Visual Feature Extraction

Multimodal detection requires relevant features to be extracted from both video and audio streams before their integration.

A. Visual Feature Extraction: Visual features describe facial appearance and movement. Common characteristics include facial texture, landmarks, eye and mouth movements, expressions, head pose and spatial or temporal inconsistencies. CNN-based and transformer-based models can extract these representations from selected frames or video sequences.

B. Audio Feature Extraction: Audio features represent acoustic characteristics of speech. Common representations include MFCCs, LFCCs, CQCCs, spectrograms, raw waveforms and learned speech representations.

C. Cross-Modal Features: Cross-modal features describe relationships between audio and video, including correspondence between mouth movements and speech activity, temporal alignment, visible speaker identity and voice characteristics, and synchronization between phonetic events and facial movements.

Combining these features can provide information that may not be available through independent audio or video analysis.

6.3 Audio-Visual Synchronization Analysis

Audio-visual synchronization analysis examines whether the timing and characteristics of an audio stream correspond appropriately with the associated video stream.

The relationship between lip movements and spoken speech is particularly relevant.

Other relevant characteristics include:

1. Timing differences between speech and mouth movements.
2. Inconsistent opening and closing of the mouth.
3. Unnatural transitions between facial movements.
4. Differences between speech activity and visible facial activity.
5. Temporal inconsistencies between audio and video streams.

However, synchronization differences do not necessarily indicate manipulation.

6.4 Multimodal Feature Fusion Techniques

Feature fusion determines how information from audio and video is combined. The main approaches include:

A. Early Fusion: Early fusion combines audio and visual features before the final classification stage. It allows joint learning of relationships between the modalities, although differences in feature dimensions may require additional preprocessing.

B. Late Fusion: Late fusion processes audio and video independently before combining their classification outputs. It supports modularity because individual detection models can be developed and updated independently while retaining modality-specific results.

C. Hybrid Fusion: Hybrid fusion combines features or representations at multiple stages, potentially integrating intermediate feature fusion with decision-level integration. It offers flexibility but may increase architectural complexity.

D. Attention-Based Fusion: Attention-based fusion uses attention mechanisms to determine the relative importance of features from different modalities. Cross-modal attention allows audio and visual representations to interact during processing.

E. Temporal Fusion: Temporal fusion combines information across multiple time intervals. It is useful when manipulation-related evidence appears only in selected portions of a recording. Temporal windows and fusion frequency must be selected carefully for real-time applications.

6.5 Deep-Learning Architectures for Multimodal Detection

Several architectures can analyse and combine audio-visual information.

A. Dual-Branch Architecture: Separate audio and video branches extract modality-specific features before combining their outputs. The video branch may use CNNs or transformers, while the audio branch may use spectrogram-based CNNs, waveform models or pretrained speech representations.

B. Cross-Attention Architecture: Cross-attention allows information from one modality to influence the representation of another. For example, audio features can attend to visual features to examine relationships between spoken speech and visible facial movements.

C. Multimodal Transformer Architecture: Multimodal transformers use attention mechanisms to integrate audio and visual representations. They can model relationships within individual modalities and interactions between them, although long sequences may increase memory usage and inference time.

D. Synchronization-Based Architecture: These approaches examine temporal relationships between audio and video, such as correspondence between acoustic speech events and visible mouth movements.

E. Hybrid Multimodal Architecture: Hybrid architectures combine techniques such as CNN-based visual extraction, transformer-based audio representation and attention-based fusion. Their complexity may make real-time deployment more challenging.

6.6 Benchmark Datasets for Multimodal Deepfake Detection

Multimodal research requires datasets containing corresponding audio and video information.

A. FakeAVCeleb: Provides manipulated audio-visual content for research involving manipulated facial content and synthetic speech [7].

B. AV-Deepfake1M: A large-scale audio-visual deepfake dataset designed to support research into multimodal manipulation detection [8].

C. LAV-DF: The Large-scale Audio-Visual Deepfake dataset supports research into localized audio-visual manipulation and temporal analysis [16].

D. Deepfake-TIMIT: Primarily associated with manipulated facial videos. Its suitability for audio-visual research depends on the availability and correspondence of the required audio information.

E. FaceForensics++: Primarily a facial manipulation video benchmark. It is relevant to visual detection but should not automatically be treated as a complete synchronized audio-visual dataset.

F. ASVspoof: These datasets are primarily intended for speech anti-spoofing research. They can provide relevant audio detection information, but their samples should not be assumed to contain corresponding video streams.

The datasets differ in modality coverage, manipulation techniques, temporal annotations and evaluation conditions.

6.7 Evaluation of Multimodal Deepfake Detection

Evaluation should consider individual modality performance as well as the integrated detection result.

A. Classification Metrics: Accuracy, precision, recall, F1-score and Area Under the Receiver Operating

Characteristic Curve (AUC) can be used to assess classification performance.

B. Modality-Specific Performance: Audio and video outputs should also be evaluated independently to identify the contribution and reliability of each modality.

C. Synchronization Evaluation: Synchronization measures can quantify temporal correspondence or alignment error, depending on the method used.

D. Temporal Localization: Where datasets provide temporal manipulation annotations, evaluation can examine whether the system identifies manipulated portions correctly.

E. Processing Latency: Evaluation should consider media acquisition, preprocessing, individual model inference, feature fusion and result delivery.

F. Modality Robustness: Evaluation should consider noisy audio, low-quality video, temporary stream interruptions and synchronization errors.

A comprehensive evaluation should therefore consider classification performance, temporal localization, modality reliability and computational efficiency rather than relying on a single metric.

6.8 Challenges in Real-Time Multimodal Detection

Real-time multimodal detection introduces additional challenges because audio and video must be processed together while maintaining acceptable response times.

A. Audio-Visual Synchronization: Network delays, buffering and encoding differences can cause the streams to arrive at different times. Ordinary communication delays must be distinguished from manipulation-related inconsistencies.

B. Computational Complexity: Processing video frames and audio segments simultaneously requires computational resources. Complex fusion architectures may further increase memory usage and inference time.

C. Unequal Modality Quality: Audio and video quality may differ significantly. A session may contain clear video but noisy audio, or clear audio with low-resolution video.

D. Missing Modalities: Some communication environments may temporarily lose audio or video information. The system should represent uncertainty rather than treating every incomplete input as evidence of manipulation.

E. Temporal Window Selection: Short analysis windows may provide limited context, while longer windows may increase detection delay.

F. Generalization: Models may perform differently when exposed to unseen generation techniques, recording environments or communication platforms.

G. Privacy and Data Management: Audio and video recordings may contain sensitive personal information. A practical architecture should consider consent, access control, secure transmission, limited data retention and privacy-aware processing.

6.9 Proposed Unified Audio-Visual Detection Framework

This study proposes a conceptual unified detection framework that combines independent audio and video detection services with a multimodal integration component.

The proposed stages are:

1. Media Acquisition: Receive authorized audio and video inputs from a communication application or controlled media source.
2. **Stream Separation:** Organize the incoming content into corresponding audio and video streams.
3. **Audio Processing:** Perform decoding, segmentation, preprocessing, feature extraction and synthetic speech detection.
4. **Video Processing:** Perform frame extraction, preprocessing, visual feature extraction and video manipulation detection.
5. **Synchronization Analysis:** Examine temporal correspondence between the processed audio and video streams.
6. **Multimodal Integration:** Combine individual detection outputs and synchronization-related information using a defined fusion strategy.
7. **Decision Generation:** Produce an overall result with relevant modality-specific information.
8. **Application Communication:** Transmit the result to the Yii2 backend through an authenticated API.
9. **Result Management:** Store relevant request metadata and detection outputs in MySQL.

The conceptual processing flow is:

Audio Input → Audio Detection Service

Video Input → Video Detection Service

Audio Output + Video Output + Synchronization Information → Multimodal Integration → Overall Detection Result → Yii2 Backend → MySQL Database and Dashboard.

The architecture keeps audio and video inference independent while providing a separate component for combining their outputs.

The framework is a conceptual architecture.

6.10 Conceptual Decision-Fusion Strategy

A decision-fusion strategy can combine outputs from the audio and video detection services with synchronization-related information.

Let:

- S_v represent the video manipulation score.
- S_a represent the audio manipulation score.

- S_s represent the synchronization-related score.
- w_v, w_a and w_s represent the corresponding fusion weights.

A conceptual combined score can be expressed as:

$$S_{av} = w_v S_v + w_a S_a + w_s S_s \quad (1)$$

Subject to:

$$w_v + w_a + w_s = 1 \quad (2)$$

The weights represent the relative contribution of each component.

The combined score should not automatically be interpreted as a calibrated probability.

The proposed framework may consider three possible outcomes:

A. Genuine: Available audio, video and synchronization evidence does not provide sufficient indication of manipulation.

B. Manipulated: Combined evidence indicates characteristics associated with manipulation.

C. Inconclusive: Available information is insufficient or unreliable for confident classification. This may occur when a modality is missing, stream quality is poor or detection outputs conflict.

The inconclusive category avoids forcing a definitive classification when the available evidence is inadequate.

6.11 Discussion

Multimodal audio-visual detection provides an opportunity to examine manipulated multimedia through complementary visual and acoustic evidence.

Early fusion supports joint feature learning, while late fusion offers greater modularity.

The proposed framework addresses these considerations through separate audio and video inference services followed by a multimodal integration component.

6.12 Section Summary

This section examined multimodal audio-visual deepfake detection, including visual and acoustic feature extraction, synchronization analysis, feature-fusion techniques and deep-learning architectures.

Finally, a conceptual unified detection framework and decision-fusion strategy were presented.

VII. REAL-TIME DEEPFAKE DETECTION: TECHNICAL CHALLENGES AND PERFORMANCE CONSIDERATIONS

7.1 Real-Time Deepfake Detection Architecture

Real-time deepfake detection requires continuous or repeated analysis of incoming audio and video streams while maintaining acceptable processing delays.

A real-time detection architecture generally includes the following components:

1. Media Acquisition: Receives authorized audio and video streams or controlled media references.
2. Preprocessing: Decodes and prepares incoming media for analysis.
3. Audio Detection Service: Analyses acoustic features to identify characteristics associated with synthetic or manipulated speech.
4. Video Detection Service: Analyses selected video frames and temporal information to identify visual manipulation.
5. Multimodal Integration: Combines available detection outputs and synchronization-related information.
6. Application Backend: Uses Yii2 to manage detection requests, users and application operations.
7. Database Management: Uses MySQL to store request information, processing metadata and detection results.
8. Result Presentation: Displays detection outcomes and associated information through the application interface.

The architecture separates computationally intensive AI inference from application-level management.

7.2 Processing Latency and Response Time

Processing latency represents the time required to process incoming media and produce a detection result.

$$T_{total} = T_{acquisition} + T_{preprocessing} + T_{inference} + T_{fusion} + T_{communication} \quad (3)$$

Where:

- $T_{acquisition}$ represents the time required to receive the media segment.
- $T_{preprocessing}$ represents the time required for decoding and input preparation.
- $T_{inference}$ represents the AI model processing time.
- T_{fusion} represents the time required to combine the detection outputs.
- $T_{communication}$ represents the communication and result-delivery time.

Audio and video inference may operate in parallel when the infrastructure supports it.

Important factors influencing response time include video resolution, frame sampling frequency, audio segment duration, model complexity, hardware availability, network delays and concurrent requests.

A real-time system must balance the amount of media analysed against the time required to generate a useful result.

7.3 Computational Complexity and Resource Optimization

Deep-learning models may require substantial computational resources, particularly when processing high-resolution video and continuous audio streams.

The principal resources include CPU capacity, GPU availability, system memory, video memory, storage, network bandwidth and concurrent processing capability.

Several optimization techniques may be considered:

A. Model Quantization: Reduces the numerical precision used by model parameters and operations. Depending on the architecture and hardware, it may reduce memory requirements and improve inference efficiency.

B. Model Pruning: Removes selected parameters or connections to reduce model size and potentially improve processing efficiency. Its effect on detection reliability requires evaluation.

C. Knowledge Distillation: Transfers learned information from a larger teacher model to a smaller student model, supporting the development of compact detection architectures.

D. Frame Sampling: Analyses selected video frames rather than every incoming frame, reducing processing requirements while attempting to retain relevant visual information.

E. Audio Segmentation: Divides continuous audio into manageable segments to balance acoustic context and processing delay.

F. Hardware Acceleration: Uses supported GPUs and other acceleration technologies to improve inference performance. Actual benefits depend on the architecture, hardware and software environment.

Optimization should not be judged solely by processing speed.

7.4 Network Bandwidth and Media Compression

Online communication applications transmit audio and video through networks with varying bandwidth and connection quality.

Video compression can affect facial textures, image details and blending artefacts, while audio compression can modify spectral characteristics and remove acoustic details.

Network conditions may also introduce packet loss, variable transmission delay, jitter, temporary stream interruption and audio-visual synchronization differences.

A practical detection system should therefore be evaluated under different compression levels, resolutions, audio bitrates and transmission conditions.

7.5 Generalization and Adversarial Robustness

Generalization refers to a model's ability to maintain useful detection performance when analysing media that differs from its training data.

Several factors affect robustness:

A. Cross-Dataset Generalization: Models may learn dataset-specific characteristics rather than general indicators of manipulation. Evaluation across different datasets can help identify these limitations.

B. Unseen Manipulation Techniques: New generative models may produce content with characteristics not represented in training data, reducing the reliability of existing detectors.

C. Media Transformation: Resizing, compression, filtering and background noise may alter features used by detection models.

D. Adversarial Manipulation: Deliberately modified inputs may interfere with AI-based classification and require dedicated robustness evaluation.

E. Domain Adaptation: Transfer learning and domain adaptation may help models operate in environments that differ from their original training conditions. Their effectiveness requires evaluation using suitable data.

Robustness should therefore be considered alongside conventional classification metrics.

7.6 Detection Accuracy Versus Processing Efficiency

Real-time detection involves balancing the amount of information analysed, model complexity and response time.

TABLE 7.1: DETECTION PERFORMANCE AND PROCESSING EFFICIENCY CONSIDERATIONS

Strategy	Benefit	Limit
High-res video	More facial detail	More compute and memory
Lower frame rate	Faster analysis	May miss brief edits
Long audio clips	More speech context	Slower result
Short audio clips	More frequent checks	Less speech context
Large models	Richer features	Higher inference cost
Light models	Lower resource use	May lose feature detail
Multimodal fusion	Combines audio and video	Needs sync and more compute
Parallel inference	Less waiting	Needs service coordination

The selection of an appropriate strategy depends on application requirements, target hardware, communication conditions and acceptable detection delay.

7.7 Privacy, Security and Data Protection

Audio and video recordings may contain personal information, biometric characteristics and private communication details.

The proposed architecture should consider the following measures:

A. Secure Media Transmission: Media references and application requests should use secure communication channels. Transport Layer Security (TLS) can help protect data exchanged between the client, Yii2 backend and AI services.

B. Authentication and Authorization: Only authenticated and appropriately authorized users or services should access detection operations and stored results.

C. Controlled Media Storage: Raw audio and video should not be retained by default unless storage is necessary and appropriately authorized. Where possible, the application may retain metadata and detection results rather than complete media content.

D. Access Control: Role-Based Access Control (RBAC) can restrict access to user accounts, detection requests, administrative operations and stored results.

E. Secure Database Management: MySQL should use appropriate access permissions, secure credentials, input validation and controlled database operations.

F. Audit Logging: Relevant security and application events may be recorded for monitoring, accountability and troubleshooting.

G. Retention and Deletion Policies: Appropriate retention periods should be defined for media, processing records and detection results.

H. Consent and Transparency: Users should receive clear information about media processing, detection purposes and the limitations of automated classification.

7.8 Scalability and Distributed Processing

A real-time detection application may need to handle multiple users and simultaneous communication sessions.

A scalable architecture should separate request handling, AI inference and result management.

A. Asynchronous Processing: Yii2 can submit detection requests to a processing queue rather than waiting for every inference operation to finish within the initial HTTP request.

B. Background Workers: Independent worker processes can retrieve queued tasks and communicate with the appropriate AI inference services.

C. Load Distribution: Incoming inference requests may be distributed across available processing instances according to workload and resource availability.

D. Independent Service Scaling: Audio and video services may require different computational resources. Separating them allows each service to scale according to its processing demand.

E. Database Optimization: MySQL indexing, appropriate table design and efficient query execution can help manage increasing volumes of detection requests and results.

F. Monitoring: System monitoring can track request queues, service availability, processing delays, resource consumption and application errors.

These considerations provide a basis for scaling the proposed framework.

7.9 Proposed Performance Evaluation Framework

The proposed architecture requires structured evaluation before it can be considered suitable for practical real-time deployment.

A. Detection Performance

- Accuracy, precision, recall and F1-score.
- Area Under the Receiver Operating Characteristic Curve (AUC).
- Equal Error Rate where relevant to audio anti-spoofing.

B. Processing Performance

- Model inference time.
- Preprocessing time.
- Audio-video fusion time.
- End-to-end response time.
- Throughput and memory consumption.
- CPU and GPU utilization.

C. Robustness

- Performance under different compression levels.
- Performance under background noise.
- Performance across different datasets.
- Generalization to unseen manipulation methods.
- Performance under missing or degraded modalities.

D. Application Performance

- API response time.
- Concurrent request handling.
- Queue processing time.
- Database query response time.
- Service availability and error recovery.

E. Privacy and Security

- Authentication and authorization effectiveness.
- Secure API communication.
- Media access restrictions.
- Data retention and deletion controls.
- Audit logging and monitoring.

A future evaluation may use suitable benchmark datasets for model-level assessment and a controlled deployment environment for application-level performance testing.

No experimental measurements or performance values are reported in this study because the proposed framework has not been implemented or experimentally validated.

7.10 Integration Considerations for the Yii2–MySQL Framework

The proposed Yii2–MySQL architecture manages detection requests and results while delegating computationally intensive media analysis to independent AI services.

A. Yii2 Application Layer: Yii2 handles user authentication, request management, API endpoints, access control and result presentation.

B. AI Inference Layer: An independent machine-learning service performs audio, video and multimodal inference. A Python-based environment using frameworks such as PyTorch or TensorFlow may be considered.

C. REST API Communication: The Yii2 backend communicates with AI services through authenticated Application Programming Interfaces (APIs), supporting request submission, status retrieval and result delivery.

D. MySQL Database Layer: MySQL stores structured application information, including user records, detection requests, processing status, model information and detection results.

E. Asynchronous Task Management: A queue-based mechanism may manage long-running detection tasks and prevent the web application from remaining blocked during AI inference.

F. Result Presentation: The application dashboard may display detection status, analysed modality, classification output, confidence information where available, processing duration and model version.

The separation of these components supports modular development and allows the application layer to remain independent of the internal implementation of the AI models.

7.11 Discussion

Real-time deepfake detection requires coordination among several technical components.

Audio and video processing introduce different computational requirements, while multimodal integration adds synchronization and feature-fusion overhead.

The proposed framework addresses these considerations through independent AI inference services, a Yii2 application backend and a MySQL database.

However, the framework remains conceptual.

7.12 Section Summary

This section examined the technical challenges associated with real-time audio, video and multimodal deepfake detection.

A qualitative comparison of processing strategies and a proposed performance evaluation framework were also presented.

These considerations provide the technical foundation for the proposed application architecture presented in the next section.

VIII. PROPOSED YII2–MYSQL-BASED APPLICATION FRAMEWORK

8.1 System Requirements and Proposed Functionalities

The proposed framework is designed to provide a web-based environment for managing AI-driven audio, video and multimodal deepfake detection requests.

The framework uses the Yii2 PHP framework for backend application management and MySQL for structured data storage.

The proposed system considers the following functional requirements:

1. User Management: Registration, authentication and management of application users.
2. Media Input Management: Submission of authorized audio and video inputs or controlled references to communication streams.
3. Detection Request Management: Creation, tracking and management of detection requests.
4. Audio Detection Integration: Communication with an external AI service for synthetic speech and audio manipulation detection.
5. Video Detection Integration: Communication with an external AI service for facial and video manipulation detection.
6. Multimodal Analysis: Integration of audio, video and synchronization-related detection outputs.
7. Result Management: Storage and retrieval of detection results and associated processing information.
8. Dashboard and Reporting: Presentation of detection requests, processing status and classification outcomes.
9. Administrative Management: Monitoring of users, detection activities and application operations.
10. Security and Privacy: Protection of user information, media references, API communication and stored results.

These functionalities represent proposed system requirements and have not been presented as implemented features.

8.2 Overall Application Architecture

The proposed application follows a modular, service-oriented architecture in which the web application, database and AI inference services perform separate responsibilities.

The major architectural components are described below.

A. Presentation Layer

The presentation layer provides the user interface through which authorized users interact with the application.

B. Yii2 Application Layer

The Yii2 framework manages application logic, user authentication, access control, request validation, REST API endpoints and communication with external AI services.

C. AI Inference Layer

The AI inference layer contains independent services for video deepfake detection, audio deepfake detection and multimodal result integration.

D. Task Processing Layer

A background task-processing mechanism may manage long-running detection requests.

E. Database Layer

The MySQL database stores structured information such as user records, detection sessions, media request metadata, model information and detection results.

F. Reporting Layer

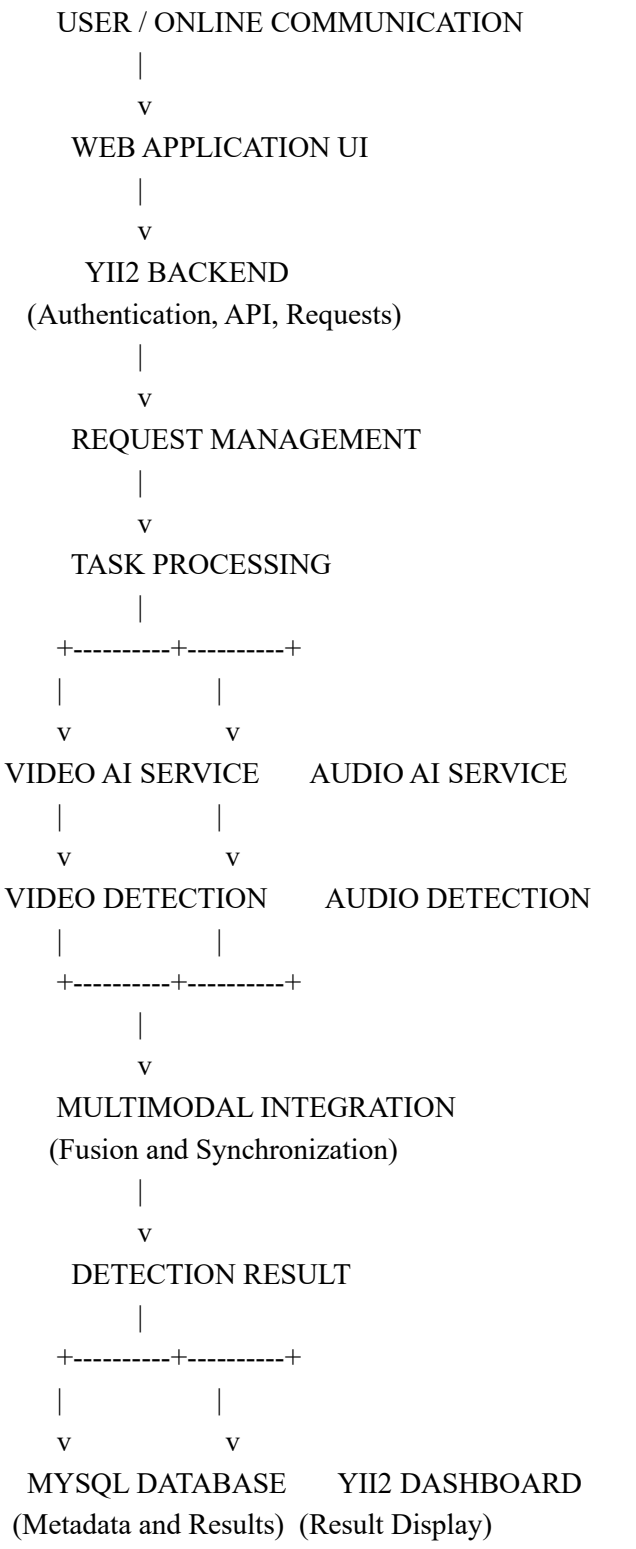
The reporting layer presents detection outcomes and application-level information through the web interface.

The separation of these components allows the AI models to be modified or replaced independently of the Yii2 application and database structure.

8.3 Proposed System Architecture Diagram

The proposed architecture can be represented through the following conceptual component flow.

Figure 8.1: Proposed Yii2–MySQL-Based Deepfake Detection Architecture



The diagram illustrates the logical separation between application management and AI-based inference.

The Yii2 backend manages requests and communicates with the inference services.

The architecture is conceptual and represents the proposed design rather than a deployed software system.

8.4 Yii2 Backend Architecture

Yii2 is a PHP-based web application framework that provides facilities for developing structured, database-driven applications and RESTful services.

In the proposed system, Yii2 serves as the central application management layer.

The main Yii2 components are described below.

A. Controllers

Controllers manage incoming application requests and coordinate the required application operations. Proposed controllers may include:

- UserController for user-related operations.
- DetectionController for detection request management.
- SessionController for communication session information.
- ResultController for detection result retrieval.
- AdminController for administrative functions.

B. Models

Yii2 models represent application data and business rules.

C. Views

Views provide the user interface for authentication, media submission, detection status, result presentation and administrative reporting.

D. REST API Endpoints

The backend exposes controlled API endpoints for receiving detection requests, retrieving request status and presenting completed results.

E. Authentication and Access Control

Yii2 authentication and authorization facilities can be used to restrict application operations according to user roles and permissions.

F. Application Services

Separate application service classes may handle communication with the external AI inference services.

The proposed backend structure supports modular application development and provides a clear boundary between web application operations and machine-learning inference.

8.5 REST API Integration with AI Detection Services

The Yii2 backend communicates with the external AI inference services through RESTful APIs.

The communication process may involve the following operations:

1. The user submits an authorized media input or controlled stream reference.
2. The Yii2 backend validates the request and creates a detection record.
3. The backend submits the required processing task to the appropriate AI service.

4. The AI service performs audio, video or multimodal analysis.
5. The inference service returns a processing status or detection result.
6. The Yii2 backend updates the corresponding request record.
7. The application presents the result to the authorized user.

A conceptual API request may be represented as follows:

Example 8.1: Detection Request

```
{
 "request_id": "REQ-1001",
 "session_id": "SESSION-2001",
 "media_type": "audio_video",
 "audio_reference": "controlled-audio-reference",
 "video_reference": "controlled-video-reference",
 "analysis_type": "multimodal"
}
```

The values are illustrative and do not represent actual application records.

A conceptual response may be represented as follows:

Example 8.2: Detection Response

```
{
 "request_id": "REQ-1001",
 "status": "completed",
 "audio_result": "genuine",
 "video_result": "manipulated",
 "overall_result": "inconclusive",
 "model_version": "example-version",
 "processing_time_ms": 420
}
```

The response values, including the processing time, are illustrative only.

For longer processing tasks, the backend may initially receive a request identifier and processing status.

For continuous communication, the architecture should support controlled segment-based processing or secure stream references rather than assuming that an entire live stream will be uploaded through a conventional PHP request.

8.6 MySQL Database Architecture

MySQL is proposed as the relational database management system for storing structured application information.

The proposed database may contain the following tables.

A. Users Table

Stores user account information and authentication-related details.

Suggested fields:

- user_id
- username
- email
- password_hash
- role
- account_status
- created_at

B. Detection Sessions Table

Stores information about communication or detection sessions.

Suggested fields:

- session_id
- user_id
- session_type
- session_status
- started_at
- ended_at

C. Media Requests Table

Stores metadata associated with submitted detection requests.

Suggested fields:

- request_id
- user_id
- session_id
- media_type
- audio_reference
- video_reference
- analysis_type
- request_status
- created_at

D. Detection Results Table

Stores the outputs generated by the AI inference services.

Suggested fields:

- result_id
- request_id
- audio_result
- video_result
- multimodal_result
- confidence_information
- processing_time
- model_version
- completed_at

E. Model Registry Table

Stores information about the detection models available to the application.

Suggested fields:

- model_id
- model_name
- model_type
- model_version
- deployment_status

- created_at

F. Detection Logs Table

Stores application-level information about detection processing and status changes.

Suggested fields:

- log_id
- request_id
- service_name
- processing_status
- event_description
- created_at

G. Security Audit Logs Table

Stores relevant security and administrative events.

Suggested fields:

- audit_id
- user_id
- action_type
- event_description
- created_at

The proposed relationships between these tables allow user records to be associated with detection sessions, requests and results.

The database design is conceptual.

Raw media storage is not assumed by default.

8.7 Detection Request and Result Management Workflow

The proposed detection workflow is designed to manage requests from initial submission through AI inference and result presentation.

The workflow consists of the following stages.

Stage 1: User Authentication

The user accesses the web application and completes the required authentication process.

Stage 2: Media Submission

The authorized user submits an audio or video input, or provides a controlled reference to an eligible communication stream.

Stage 3: Request Validation

The Yii2 backend validates the request parameters, user permissions, media type and other application-level requirements.

Stage 4: Request Registration

A new detection request is created and associated with the relevant user and session information.

Stage 5: Task Submission

The request is forwarded to the appropriate processing queue or AI inference service.

Stage 6: AI-Based Analysis

The audio and video services process their respective inputs.

Stage 7: Result Generation

The inference services generate the classification outputs and associated processing information.

Stage 8: Database Update

The Yii2 backend records the result information and updates the processing status in MySQL.

Stage 9: Result Presentation

The application dashboard displays the detection outcome and relevant request information to the authorized user.

The conceptual workflow is:

User Authentication → Media Submission → Request Validation → Request Registration → AI Processing → Multimodal Integration → Result Storage → Dashboard Presentation.

For continuous communication, this workflow may operate repeatedly over selected media segments rather than treating the entire communication session as one completed file-processing task.

8.8 Authentication and Application Security

Security is an essential requirement because the proposed application handles user accounts, media references and detection results.

The following security measures are considered in the proposed architecture.

A. User Authentication

The application should provide secure authentication mechanisms and appropriate password-hashing practices.

B. Role-Based Access Control

Role-Based Access Control (RBAC) can restrict operations according to user permissions.

C. API Authentication

Communication between Yii2 and external AI services should use authenticated API requests.

D. Secure Communication

Transport Layer Security should be used to protect communication between the browser, application backend and inference services.

E. Input Validation

All incoming requests should be validated to prevent malformed data, unauthorized operations and unexpected input values.

F. Media Validation

Uploaded media should be checked for permitted file types, file size and other relevant restrictions.

G. Database Security

Database access should follow the principle of least privilege.

H. Audit Logging

Important application and administrative operations should be recorded for monitoring and accountability.

I. Privacy Protection

Media retention, user consent, access restrictions and deletion procedures should be defined according to the application requirements.

These measures are proposed security requirements.

8.9 Proposed Dashboard and Reporting Functionalities

The proposed dashboard provides a centralized interface for managing detection requests and viewing their results.

The following functionalities may be included.

A. User Dashboard

Displays the user's recent detection requests, current processing status and previously completed results.

B. Media Submission Interface

Provides an interface for submitting authorized audio and video inputs or selecting an eligible analysis type.

C. Detection Status Monitoring

Displays request states such as pending, processing, completed, failed or inconclusive.

D. Detection Result Display

Presents the audio result, video result and combined multimodal outcome where available.

E. Processing Information

Displays relevant details such as request identifiers, processing duration, model version and available confidence information.

F. Historical Records

Allows authorized users to review previous detection requests and results according to the application's retention policy.

G. Administrative Dashboard

Provides administrative information about users, processing activity, model availability and application status.

H. Reporting

Allows the generation of summaries of detection activity and application-level processing information.

The dashboard is intended to present model outputs transparently.

8.10 Advantages and Limitations of the Proposed Framework

The proposed Yii2–MySQL-based architecture offers several potential advantages.

A. Modular Architecture

The separation of application management, database operations and AI inference supports independent development and maintenance.

B. Independent AI Services

Audio and video detection models can be updated or replaced without requiring major changes to the Yii2 backend.

C. Centralized Request Management

The application provides a common environment for managing users, sessions, detection requests and results.

D. Structured Data Storage

MySQL supports relational storage of application records and detection metadata.

E. Future Scalability

Independent inference services and asynchronous task processing provide opportunities for scaling different components according to workload.

F. Multimodal Integration

The architecture provides a common structure for combining audio and video detection outputs.

However, the proposed framework also has limitations.

1. Its actual detection performance depends on the quality and reliability of the external AI models.
2. Real-time operation requires appropriate hardware, network infrastructure and processing optimization.
3. Multimodal integration introduces additional synchronization and computational requirements.
4. The conceptual database structure requires further refinement during implementation.
5. Security, privacy and scalability must be evaluated under realistic deployment conditions.
6. The architecture has not been implemented or experimentally validated in the present study.

Therefore, the framework should be understood as a proposed software architecture rather than a completed deepfake detection application.

8.11 Discussion

The proposed framework addresses the application-level requirements identified through the review of existing audio, video and multimodal deepfake detection techniques.

Yii2 is assigned responsibility for authentication, request management, API communication and result presentation.

This architecture provides a modular foundation for future implementation.

However, the effectiveness of the proposed framework cannot be established through architectural design alone.

8.12 Section Summary

This section presented a conceptual Yii2–MySQL-based application framework for integrating AI-driven audio, video and multimodal deepfake detection.

The section also discussed detection request management, authentication, application security, dashboard functionality and the advantages and limitations of the proposed architecture.

The framework separates AI inference from web application management and provides a structured foundation for future implementation.

IX. COMPARATIVE DISCUSSION AND RESEARCH GAPS

9.1 Comparative Analysis of Video Deepfake Detection Techniques

Video deepfake detection methods differ in their feature extraction strategies and ability to analyse spatial and temporal information.

Table 9.1 compares the major approaches in terms of their features, advantages and limitations.

TABLE 9.1: COMPARISON OF VIDEO DEEPFAKE DETECTION METHODS

Method	Main features	Strength	Limitation
CNN (Xception, ResNet)	Facial texture; blending; spatial cues	Strong frame features	Weak temporal context
EfficientNet	Scaled spatial features	Flexible model size	Needs tuning for speed
RNN / LSTM / GRU	Changes across frames	Models time patterns	Sequence adds delay
3D CNN	Space and time features	Joint clip analysis	Higher compute cost
Vision Transformer	Patch and frame relations	Broad context	High memory demand
Frequency analysis	FFT / DCT patterns	Adds spectral cues	Sensitive to compression
Artifact analysis	Edges; light; skin; geometry	Useful visual clues	Artifacts may be absent

The comparison indicates that each approach addresses different aspects of visual manipulation.

9.2 Comparative Analysis of Audio Deepfake Detection Techniques

Audio deepfake detection approaches differ according to their acoustic representations and learning architectures.

Table 9.2 summarizes the major characteristics of these approaches.

TABLE 9.2: COMPARISON OF AUDIO DEEPFAKE DETECTION METHODS

Method	Input / feature	Strength	Limitation
MFCC	Mel cepstral features	Compact speech features	May miss advanced synthesis
LFCC	Linear cepstral features	Used in spoof detection	Recording-sensitive
CQCC	Constant-Q features	Alternative spectral view	Task-dependent
CNN	Spectrograms	Learns local patterns	Depends on training data
RNN / LSTM	Acoustic sequences	Models speech over time	Longer inference
RawNet2	Raw waveform	Learns from audio directly	Needs suitable data and compute
AASIST	Spectro-temporal graph	Models linked features	More complex architecture
Transformer	Attention-based features	Captures broad context	Higher compute demand

Audio detection therefore includes both handcrafted acoustic features and learned neural representations.

9.3 Comparison of Unimodal and Multimodal Detection Approaches

Unimodal detection examines a single media source, such as video or audio, whereas multimodal detection combines information from multiple sources.

Table 9.3 presents a qualitative comparison.

TABLE 9.3: UNIMODAL AND MULTIMODAL DETECTION

Factor	Video-only	Audio-only	Multimodal
Input	Video frames	Audio segments	Audio and video
Main cues	Face; motion; texture	Voice; spectrum; rhythm	Visual, acoustic and cross-modal cues
Processing	Frame analysis	Speech analysis	Joint or parallel analysis
Sync analysis	Not required	Not required	Important
Missing input	Video unavailable	Audio unavailable	Must handle missing stream
Key benefit	Visual evidence	Speech evidence	Combines evidence
Key challenge	Misses audio edits	Misses visual edits	More compute and sync needs

Multimodal detection can combine complementary evidence, but its effectiveness depends on the availability and quality of both modalities.

9.4 Comparative Analysis of Benchmark Datasets

Benchmark datasets support the evaluation of deepfake detection methods, but they differ in media type, manipulation techniques, recording conditions and research objectives.

Table 9.4 summarizes the datasets considered in this review.

TABLE 9.4: OVERVIEW OF DEEPFAKE DETECTION DATASETS

Dataset	Modality	Main use
FaceForensics++	Video	Facial manipulation and visual artifacts
Celeb-DF	Video	Realistic face-swap detection
DFDC	Video	Large-scale video forgery research
DeeperForensics-1.0	Video	Robustness to visual changes
ASVspoof 2019	Audio	Synthetic, converted and replayed speech
ASVspoof 2021	Audio	Speech spoofing in varied conditions
WaveFake	Audio	Synthetic speech detection
FoR	Audio	Genuine vs synthetic speech
FakeAVCeleb	Audio-visual	Face and speech manipulation
AV-Deepfake1M	Audio-visual	Large-scale multimodal detection
LAV-DF	Audio-visual	Localized audio-visual forgery

Dataset selection should reflect the intended detection objective.

9.5 Accuracy, Latency and Computational Trade-Offs

Detection performance and processing efficiency are closely related in real-time applications.

Reducing video frame sampling or shortening audio segments may also improve processing efficiency, although it can limit the information available to the detection model.

Table 9.5 summarizes these trade-offs.

TABLE 9.5: ACCURACY AND PROCESSING TRADE-OFFS

Strategy	Potential benefit	Possible limitation
High-resolution video	More facial detail	Higher compute and memory use
Lower frame rate	Faster processing	May miss brief edits
Long audio segments	More speech context	Longer response time
Short audio segments	Faster repeated checks	Less acoustic context
Large neural models	Richer feature learning	Higher inference cost
Lightweight models	Lower resource use	May lose feature detail
Multimodal fusion	Combines audio and video cues	Needs sync and extra compute
Parallel inference	Can reduce waiting time	Needs service coordination

A complete evaluation should consider accuracy, precision, recall, F1-score, AUC, EER where applicable, inference latency, end-to-end response time, throughput and resource consumption.

Numerical comparisons should be made only when datasets, evaluation protocols and experimental conditions are sufficiently comparable.

9.6 Comparative Analysis of Existing Research and the Proposed Framework

Existing research has contributed to video, audio and multimodal detection through improvements in detection architectures, feature representations and benchmark evaluation.

Table 9.6 compares the general focus of existing detection research with the proposed framework.

TABLE 9.6: EXISTING RESEARCH AND PROPOSED FRAMEWORK

Aspect	Existing research focus	Proposed framework
Main focus	Detection methods and evaluation	Application integration
Video analysis	CNN, temporal and transformer models	Independent video AI service
Audio analysis	Acoustic and neural models	Independent audio AI service
Multimodal analysis	Fusion and synchronization	Separate fusion component
Backend	Varies by study	Yii2 request and user management
Data storage	Study-specific	MySQL metadata and results
API integration	Not always central	Authenticated service APIs
Validation	Varies by publication	Conceptual; not yet validated

The proposed framework does not claim to introduce a new deep-learning detection algorithm.

9.7 Identified Research Gaps

The literature review and comparative analysis identify several research gaps relevant to real-time audio-visual deepfake detection.

A. Generalization to Unseen Manipulation Techniques

Detection models may behave differently when analysing media generated by techniques not represented in their training datasets.

B. Real-Time Processing Constraints

Maintaining acceptable processing delays while analysing continuous audio and video streams requires computational efficiency and suitable system resources.

C. Audio-Visual Integration

Video and audio detection are frequently investigated independently.

D. Synchronization and Temporal Consistency

Audio-visual synchronization can provide useful evidence, but network delays, buffering and natural variations in speech and facial movements complicate its interpretation.

E. Robustness Under Media Transformation

Compression, resizing, filtering, noise and transmission effects can alter the characteristics used by detection models.

F. Missing or Degraded Modalities

Multimodal systems may encounter unavailable or low-quality streams.

G. Standardized Evaluation

Differences in datasets, preprocessing methods, manipulation techniques and evaluation protocols make direct comparisons difficult.

H. Explainability and Result Interpretation

Detection systems should provide meaningful information about their outputs.

I. Privacy-Aware Deployment

Audio and video streams may contain sensitive personal information.

J. Application-Level Integration

Modular software architectures can connect independent AI detection services with web applications for request management, result storage, monitoring and reporting.

9.8 Future Research Directions

The identified gaps suggest several opportunities for future research.

A. Lightweight and Efficient Detection Models

Future studies may investigate compact architectures, quantization, pruning and knowledge distillation to reduce computational requirements while maintaining acceptable detection reliability.

B. Improved Multimodal Learning

Further research may examine cross-attention, temporal fusion and adaptive decision strategies for combining audio and video evidence under different communication conditions.

C. Generalization Across Generative Models

Training and evaluation across diverse datasets and previously unseen manipulation techniques may help improve detection robustness.

D. Real-Time Streaming Evaluation

Future studies should evaluate detection systems using continuous or segment-based audio and video streams rather than relying exclusively on offline media files.

E. Modality-Aware Decision Making

Adaptive fusion approaches may assign different importance to audio and video information according to stream quality, availability and detection reliability.

F. Explainable Deepfake Detection

Future systems may provide interpretable evidence, such as suspicious temporal intervals, visual regions or acoustic characteristics, to support human understanding of detection outputs.

G. Privacy-Preserving Detection

Research may explore data minimization, controlled processing, secure communication and other privacy-aware methods for multimedia analysis.

H. Application Implementation and Evaluation

The proposed Yii2–MySQL architecture may be implemented and integrated with suitable audio and video inference services.

I. Standardized Performance Evaluation

Future research should establish procedures that consider detection accuracy, latency, resource consumption, robustness and generalization under clearly documented experimental conditions.

9.9 Overall Discussion

The comparative analysis demonstrates that video, audio and multimodal detection approaches address different aspects of deepfake manipulation.

Multimodal detection extends individual modality analysis by examining audio-visual correspondence and combining complementary evidence.

Practical deployment therefore requires attention to latency, hardware resources, network conditions, privacy and scalability.

The framework complements existing detection research but does not replace the need for validated AI models or experimental testing.

9.10 Section Summary

This section compared video, audio and multimodal deepfake detection approaches, including their feature representations, architectures, benchmark datasets and performance considerations.

The analysis identified research gaps involving generalization, real-time efficiency, multimodal integration, synchronization, robustness, explainability, privacy and application-level deployment.

These findings provide the basis for the conclusion and future scope presented in the next section.

X. CONCLUSION AND FUTURE SCOPE

10.1 Summary of the Research

The advancement of artificial intelligence and generative technologies has increased the realism and accessibility of manipulated audio and video content, creating challenges for multimedia authenticity, digital communication and information security.

This study presented a literature-based review of AI techniques for detecting video deepfakes, synthetic speech and multimodal audio-visual manipulation.

The review also considered benchmark datasets and evaluation measures, including accuracy, precision, recall, F1-score, AUC and EER, together with processing latency and computational requirements.

The study further examined the technical requirements of real-time detection in online communication, including continuous media processing, computational efficiency, network conditions, privacy, security and scalability.

Based on these requirements, a conceptual web-based application framework was proposed using the Yii2 PHP framework and MySQL database.

10.2 Major Contributions of the Study

The major contributions are summarized as follows:

1. Review of Video Deepfake Detection: Examined CNN-based, temporal, transformer-based and frequency-domain approaches for identifying manipulated video content.
2. Review of Audio Deepfake Detection: Analysed acoustic feature extraction methods and deep-learning architectures for identifying synthetic speech and manipulated audio.
3. Multimodal Detection Analysis: Discussed audio-visual feature extraction, synchronization analysis and fusion strategies for combining audio and video evidence.
4. Performance and Deployment Considerations: Examined latency, computational complexity, media compression, generalization, robustness, privacy and scalability in real-time environments.
5. Conceptual Application Architecture: Proposed a modular Yii2–MySQL-based framework for managing detection requests, integrating independent AI inference services and storing structured results.
6. Identification of Research Gaps: Identified further research opportunities in real-time processing, multimodal integration, model generalization, explainability and privacy-aware deployment.

These contributions provide a consolidated view of the technical and application-level considerations involved in real-time audio-visual deepfake detection.

10.3 Significance of the Proposed Yii2–MySQL Framework

The proposed framework provides an application-level architecture for integrating AI-based audio, video and multimodal detection services into a web-based environment.

Yii2 manages user authentication, detection requests, API communication and result presentation, while MySQL stores user information, request metadata, processing details, model information and detection results.

The architecture also considers asynchronous task processing, controlled API communication, result monitoring and modular service management.

The practical effectiveness of the framework must be established through implementation and evaluation.

10.4 Limitations of the Study

This study has several limitations.

First, it is based on a literature review and does not include the development or experimental validation of a new deepfake detection model.

Second, the proposed Yii2–MySQL framework is a conceptual software architecture.

Third, the reviewed studies use different datasets, preprocessing methods, manipulation techniques and evaluation protocols.

Fourth, the rapid evolution of generative AI may introduce manipulation techniques that are not adequately represented in existing benchmark datasets or published detection research.

Fifth, practical real-time multimodal detection depends on hardware availability, network conditions, media quality and synchronization.

Finally, the proposed decision-fusion strategy and application workflows remain conceptual.

10.5 Future Scope

Several directions can extend the present study.

A. Implementation of the Proposed Framework

Future work may involve developing the Yii2–MySQL application and integrating it with independent AI-based audio and video detection services.

B. Integration of Advanced Detection Models

CNN, transformer-based, graph-attention and pretrained speech architectures may be evaluated and integrated according to their suitability for the target detection tasks.

C. Real-Time Streaming Support

Future development may investigate continuous audio and video processing through controlled streaming or segment-based analysis, with particular attention to buffering, synchronization and end-to-end response time.

D. Lightweight Model Development

Quantization, pruning, knowledge distillation and efficient feature extraction may help reduce computational requirements while maintaining acceptable detection reliability.

E. Improved Multimodal Fusion

Further research may examine adaptive fusion strategies that consider the reliability and availability of individual modalities, including noisy audio, degraded video and missing streams.

F. Robustness and Generalization Evaluation

Detection models may be evaluated across multiple benchmark datasets and previously unseen manipulation techniques to investigate their ability to generalize beyond their original training conditions.

G. Explainable Detection

Future systems may incorporate interpretable information, such as suspicious facial regions, acoustic characteristics and temporal intervals, to help users understand detection outputs and their limitations.

H. Privacy-Aware Processing

Further research may investigate secure media handling, data minimization, controlled retention and privacy-preserving processing techniques for online communication environments.

I. Experimental Performance Evaluation

A future implementation may be evaluated using suitable benchmark datasets and controlled communication scenarios.

These directions may help transform the proposed conceptual architecture into a practical and experimentally evaluated deepfake detection application.

10.6 Concluding Remarks

AI-based deepfake detection is an evolving research area that requires continued development of reliable detection techniques and practical deployment architectures.

This study reviewed the major technical approaches and examined the challenges associated with integrating them into online communication environments.

The proposed architecture provides a foundation for future software development and experimental investigation.

Overall, the study highlights the importance of combining advances in AI-based multimedia analysis with modular software architecture to support reliable, efficient and privacy-aware deepfake detection in online communication.

ACKNOWLEDGEMENT

The author would like to express their sincere gratitude to all researchers and scholars whose published works have contributed to the development of this study.

REFERENCES

[1] I. J. Goodfellow et al., “Generative Adversarial Nets,” in *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[2] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, “FaceForensics++: Learning to Detect Manipulated Facial Images,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1–11.

[3] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, “Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3204–3213. doi: 10.1109/CVPR42600.2020.00327.

[4] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, “DeeperForensics-1.0: A Large-Scale Dataset for Real-World Face Forgery Detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2889–2898.

[5] X. Wang et al., “ASVspoof 2019: A Large-Scale Public Database of Synthesized, Converted and Replayed Speech,” *Computer Speech & Language*, vol. 64, Art. no. 101114, 2020. doi: 10.1016/j.csl.2020.101114.

[6] J. Jung, H. Heo, H. Tak, H. Shim, J. S. Chung, B. J. Lee, H. J. Yu, and N. Evans, “AASIST: Audio Anti-Spoofing Using Integrated Spectro-Temporal Graph Attention Networks,” in *Proceedings of the IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 6367–6371. doi: 10.1109/ICASSP43922.2022.9747766.

[7] H. Khalid, S. Tariq, M. Kim, and S. S. Woo, "FakeAVCeleb: A Novel Audio-Video Multimodal Deepfake Dataset," in *Proceedings of the NeurIPS 2021 Datasets and Benchmarks Track*, 2021. Available: https://openreview.net/forum?id=TAXFsg6ZaOl.

[8] Z. Cai, S. Ghosh, A. P. Adatia, M. Hayat, A. Dhall, T. Gedeon, and K. Stefanov, "AV-Deepfake1M: A Large-Scale LLM-Driven Audio-Visual Deepfake Dataset," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 7414–7423. doi: 10.1145/3664647.3680795.

[9] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1251–1258. doi: 10.1109/CVPR.2017.195.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[11] A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.

[12] A. Dosovitskiy et al., "An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[13] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-End Anti-Spoofing with RawNet2," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[14] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[15] S. Chen et al., "WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022. doi: 10.1109/JSTSP.2022.3188113.

[16] A. Haliassos et al., "Do You Really Mean That? Content Driven Audio-Visual Deepfake Dataset and Multimodal Method for Temporal Forgery Localization," in *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2022. doi: 10.1109/DICTA56598.2022.10034605.

[17] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. Canton Ferrer, "The DeepFake Detection Challenge Dataset," arXiv preprint arXiv:2006.07397, 2020.

[18] J. Frank and L. Schönherr, "WaveFake: A Data Set to Facilitate Audio Deepfake Detection," arXiv preprint arXiv:2111.02813, 2021.

[19] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch, and K. A. Lee, "ASVspoof 2021: Towards Spoofed and Deepfake Speech Detection in the Wild," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 31, pp. 2507–2522, 2023. doi: 10.1109/TASLP.2023.3285283.

[20] R. Reimao and V. Tzerpos, "FoR: A Dataset for Synthetic Speech Detection," in Proceedings of the 2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), pp. 1–10, 2019. doi: 10.1109/SPED.2019.8906599.