National College of Ireland

Design & Evaluation of a Multi-Modal Deepfake Detection Framework for Advanced Phishing Threats

MSc Research Project
MSCCYBE_JANO24_O

Gavin Corr
Student ID: x18382836

School of Computing
National College of Ireland

Supervisor: Raza Ul Mustafa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: …….Gavin Corr……

Student ID: ……x18382836……..……

Programme: …… MSCCYBE_JANO24_O…… **Year:** …2025……..

Module: …… MSc (Research) Practicum Part 2……..……

Supervisor: …… Raza Ul Mustafa……..……

Submission Due Date: ……11/12/2025……..……

Project Title: Design & Evaluation of a Multi-Modal Deepfake Detection Framework for Advanced Phishing Threats

Word Count: ……5887 (Excluding Referecnes…… **Page Count** 19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Gavin Corr

Date: ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

Your Name/Student Number	Course	Date
Gavin Corr/x18382836	MSCCYBE_JANO24_O	11/12/2025

1 AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool
ChatGPT	Used for research and spell/Grammar checking	https://chatgpt.com/
Grammarly	Used for Grammar and Spelling	https://app.grammarly.com/

2 Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

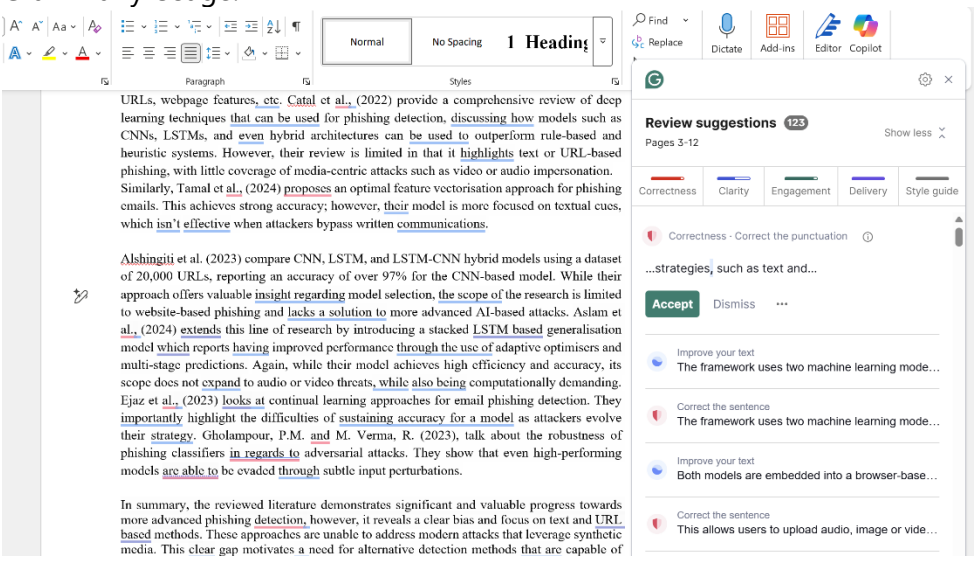
[ChatGPT]	
Used to assist in researching academic papers	
Sample Prompt: Can you show me papers related to multimodal deepfake analysis?	"Here's a curated list of multimodal deepfake analysis & detection papers you can dig into…" ChatGPT proceeded to list different papers along with their associated links.

3 Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

4 Additional Evidence:

Grammarly Usage:



Design & Evaluation of a Multi-Modal Deepfake Detection Framework for Advanced Phishing Threats

Gavin Corr
x18382836

Abstract

Traditional phishing defence strategies such as text and URL analysis, are poorly equipped to detect advanced media manipulations, such as deepfakes. Deepfake technology has made phishing attacks more sophisticated, creating a challenge for traditional detection methods. This study looks at a lightweight solution for advanced deepfake detection. The framework uses two machine learning models, a convolutional neural network (CNN) for audio deepfake detection, and a MobileNetV2 based transfer learning model for visual deepfake detection. Both models are embedded into a browser-based application deployed publicly on Hugging Face using a Gradio UI. This allows users to upload audio, image or video files for classification, confidence scoring and CSV logging. Evaluation results show that the audio model achieves a 97% accuracy rating with perfect recall scores for fake samples, which allows it to detect synthetic speech reliably. The visual model achieves 86% accuracy with high precision but lower recall on fake media, highlighting the difficulty of single-frame forgery detection and indicating potential for future work on fusion and robustness.

5 Introduction

In the cybersecurity industry, phishing remains one of the most effective and persistent cyberattacks that organisations and individuals face. Phishing attacks have traditionally leveraged deceptive tactics such as emails and fake websites; however, with the recent advancements made in artificial intelligence technology, attacks are becoming faster and more sophisticated. New deep learning techniques include generated synthetic audio, image, and videos that are more realistic than traditional media manipulation. With the initial purpose of this technology being for entertainment and creative media, deepfakes have been leveraged as a social engineering tool in order to trick individuals and organizations into compromise. Deepfake-powered synthetic identity fraud is quickly becoming a means to threaten both public trust and organisational security (Pei et al., 2024; Salvi et al., 2023; Wang et al., 2025; Tan et al., 2025). For example, Catal et al., (2022), describes how phishing attacks have gone beyond traditional static website-based attacks.

Phishing detection strategies must constantly adapt against modern threats in order to keep up with emerging threats, such as AI-powered attacks. Studies have explored audio deepfake detection (Bird & Lotfi, 2023) and facial deepfake detection (Rössler et al., 2019) in isolation, however, with a limited focus on addressing how such researched systems could be combined together, deployed, and operated within a real-world practical phishing context.
Based on current literature and limited solutions, the main goal of this research is to design and evaluate a multi-modal deepfake detection framework for advanced phishing threats.

The framework combines AI analysis of audio, image and video content, with the goal of not only classifying media as real or synthetic, but to situate deepfake detection within a practical real life security context. In order to achieve this, the project will discuss several primary objectives:

1. Implement machine learning models for audio and visual deepfake detection using CNNs and transfer learning.
2. Develop a unified, browser-based application that implements these models into a single phishing defence framework.
3. Evaluate the framework using relevant datasets and quantitative performance metrics.
4. Identify and discuss key limitations, failure modes, and operational considerations for deploying such a framework in a real-life use case.

Building on these objectives, this study is based on the following research questions:

1. **RQ1:** Using CPU-only hardware, how accurately can a CNN-based audio model trained on Mel spectrograms distinguish between real and synthetic speech?
2. **RQ2:** Under similar hardware constraints, how accurately can a MobileNetV2-based visual model trained on facial extractions detect deepfake faces in image and video media?
3. **RQ3:** Can these audio and visual models be integrated into a single lightweight browser-based application that operates at interactive speeds that are suitable for phishing defence workflows?
4. **RQ4:** What practical limitations, failure modes, and operational considerations arise after deployment of this multi-modal deepfake detection framework, and how do these findings inform recommendations for future phishing defence frameworks?

The report is structured to answer these research questions as follows. Section 2 will review related work in phishing detections and deepfake technologies. Section 3 outlines the research methodology. Section 4 presents the design specification and overall system architecture. Section 5 will discuss the implementation of model integration and application development. Section 6 evaluates the performance of both models while discussing any experimental findings. Section 7 will conclude the report with a proposition of future research and development.

6 Related Work

Based on the research questions outlined in Section 1 (RQ1-RQ4), this section will review three main strands of prior work that underpin the proposed framework. Organising the literature around these themes helps clarify what is already known about advanced phishing detection, where existing approaches fall short for deepfake-driven impersonation, and how these gaps help motivate the design choices made for the proposed framework.

6.1 Evolution of Phishing & Machine Learning-Based Detections

Traditional phishing detection research has primarily focused on analysing email content, URLs, webpage features, etc. Catal et al., (2022) provide a comprehensive review of deep learning techniques that can be used for phishing detection, discussing how models such as CNNs, LSTMs, and even hybrid architectures can be used to outperform rule-based and heuristic systems. However, their review is limited in that it highlights text or URL-based phishing, with little coverage of media-centric attacks such as video or audio impersonation.
Similarly, Tamal et al., (2024) proposes an optimal feature vectorisation approach for phishing emails. This achieves strong accuracy; however, their model is more focused on textual cues, which isn't effective when attackers bypass written communications.

Alshingiti et al. (2023) compare CNN, LSTM, and LSTM-CNN hybrid models using a dataset of 20,000 URLs, reporting an accuracy of over 97% for the CNN-based model. While their approach offers valuable insight regarding model selection, the scope of the research is limited to website-based phishing and lacks a solution to more advanced AI-based attacks. Aslam et al., (2024) extends this line of research by introducing a stacked LSTM based generalisation model which reports having improved performance through the use of adaptive optimisers and multi-stage predictions. Again, while their model achieves high efficiency and accuracy, its scope does not expand to audio or video threats, while also being computationally demanding. Ejaz et al., (2023) looks at continual learning approaches for email phishing detection. They importantly highlight the difficulties of sustaining accuracy for a model as attackers evolve their strategy. Gholampour, P.M. and M. Verma, R. (2023), talk about the robustness of phishing classifiers in regards to adversarial attacks. They show that even high-performing models are able to be evaded through subtle input perturbations.

In summary, the reviewed literature demonstrates significant and valuable progress towards more advanced phishing detection, however, it reveals a clear bias and focus on text and URL based methods. These approaches are unable to address modern attacks that leverage synthetic media. This clear gap motivates a need for alternative detection methods that are capable of analysing multiple media types.

6.2 Deepfake Generation & Detection

In parallel to the phishing research, deepfake detection has also become another focus across computer vision and speech processing communities. Rössler et al., (2019) introduced the FaceForensics++ dataset, a public and widely used benchmark dataset for the detection of manipulated facial video content. Their work showed that CNN-based and transfer-learning models are capable of identifying facial forgeries with a high accuracy rating when trained on large, labelled datasets.
Their work also highlights the need for more accessible and adequate datasets for video and image manipulation. However, their evaluation is more focused on curated forgeries rather than operational phishing media, which typically involve low quality, compressed or short videos. Consequently, their results may reduce applicability to real-world security scenarios and lacks flexibility for other media types such as audio (author's observation based on Rössler et al., 2019).

Although this study also makes use of FaceForensics++, it diverges significantly from prior research through the integration of audio, image, and video deepfake detection within a lightweight, phishing-focused, deployable framework. These elements are absent from existing studies.

For the audio detection modality, Bird and Lotfi (2023) suggest a detection framework for AI-generated speech. It is a CNN trained off of Mel-spectrograms, showing the value of spectrograms and CNN classifiers. This is a key inspiration for the use of CNN architecture and spectrograms in this research. However, their system is designed for standalone audio detection and doesn't consider the wider phishing attack workflow or integration with other media modalities.

6.3 Multi-Modal and Hybrid Approaches to Phishing and Deepfake Detection

Ige et al., (2024) proposes a multi-layer framework that is able to integrate URL and content analysis that can be used for a phishing detection workflow. An important distinction with this study is that it focuses on semantic interpretation of content (e.g. what is being said or shown) rather than explicit classifications on media as real or synthetic deepfakes. A common issue in this domain, which the authors in this study also reported, is limited dataset availability, which in turn reduces the reproducibility and generalizability of a framework.

Currently, there is lots of literature that explores multi modal approaches for deepfake detection. For example, Salvi et al., (2023), discusses a robust multimodal approach that extracts temporal audio and visual features from video media. It then exploits any inconsistencies found within or between the two media types to improve overall detection performance. Similarly, Wang et al., (2025) created ERF-BA-TFD+, which is a sophisticated fusion model that is capable of combining enhanced receptive fields with audio and visual features to achieve state of the art accuracy ratings on benchmark datasets. Additional studies such as Bekheet et al., (2024), leverages multimodal datasets such as FakeAVCeleb to train their separate audio and video models. They then combine their outputs. Jha et al., (2025) focuses on real time audio and visual deepfake detection in a video conference context, using a multimodal pipeline built around MobileNetV2. Further work, such as Tan et al., (2025), explores fusion of audio and visual modalities for multimodal deepfake detection.

6.4 Summary of findings

While the studied research clearly demonstrates value to multimodal deepfake detection, they mainly focus on architectural innovation and benchmark performance, rather than the focused development of a deployable, phishing-oriented tool for everyday use.
In contrast, this project emphasizes a more practical, usable, feasible, and reproducible solution by integrating lightweight audio, image, and video detection into a deployable browser-based application. This provides a proof-of-concept framework that forms foundational groundwork for advanced phishing defence, contrasting with the predominantly conceptual or experimentally complex multimodal models found in current research.

Altogether, the literature reviewed provides strong foundations for both static and advanced deepfake detection, however, several important questions are left unanswered in the specific context of lightweight, deployable phishing defence.

Specifically, it is still unclear how well relatively compact CNN based models or MobileNetV2 models fare under realistic CPU only constraints (RQ1 and RQ2), whether these models could be integrated into a usable browser based tool for analysts (RQ3), and what are the limitations and operational challenges when moving from conceptual, benchmark focused frameworks, to a deployed prototype (RQ4). Answering these questions forms the core focus of the methodology and evaluation of this research.

7 Research Methodology

This section goes through the research approach used to design, train, and evaluate the proposed multi-modal deepfake detection framework. In line with RQ1-RQ4, this section explores elements such as justification for datasets, technical details, model development strategy, etc., to show how the required data is obtained, processed, and analysed in order to answer the research questions introduced in Section 1.

7.1 Research Environment & Tools

The development and experimentation in this research were conducted using Python 3.9 within a Jupyter Notebook environment. Python was chosen for its flexibility, interactivity, and wide adoption in machine learning research.

Software Tools:

- **TensorFlow/Keras:** Model development
- **Librosa:** Audio processing
- **NumPy and Pandas:** Dataset manipulation, feature array construction, and evaluation logging
- **MTCNN:** Face detection
- **MobileNetV2:** Transfer learning backbone
- **OpenCV:** Video frame extraction
- **Hugging Face:** Deployment

Hardware:

- Intel i7 CPU
- 16GB RAM
- No dedicated GPU

The lightweight and limited hardware selection was used to reflect realistic hardware constraints.

7.2 Data Selection & Justification

7.2.1 Audio Dataset:

The audio dataset was sourced on Kaggle from publicly available, and MIT licensed DEEP-VOICE: DeepFake Voice Recognition Kaggle dataset (birdy654, n.d.). The dataset contains paired samples of real human speech and fake AI generated speech in audio formatting. The dataset includes several speakers and a range of synthetic voices, allowing the model to learn from diverse patterns of audio manipulation. Notably, the real-to-fake ratio was highly imbalanced with 8 real samples vs 56 fake samples, requiring a methodological intervention outlined in Section 3.3.

7.2.2 Video/Image Dataset:

For video and image training, the FaceForensics++ dataset (Rössler et al., 2019) was used. This, too, has MIT licensing with public availability. This dataset provides high-quality real videos and manipulated videos using established deepfake generation techniques. This dataset acts as a benchmark making it comparable with prior research, reproducibility and methodological consistency within the broader deepfake detection literature.

Notably, FaceForensics++ uses stage and controlled videos rather than real phishing style content. However, it currently remains the most widely accepted dataset for evaluating face manipulation in models (Rössler et al., 2019; Tan et al., 2025; Wang et al., (2025, ERF-BA-TFD+)). Its usage in this research allows for foundational model development, however, its limitations are acknowledged and addressed in Section 7.

7.3 Data Preprocessing

7.3.1 Audio Model:

1. **Segmentation & Oversampling**
   Due to the class imbalance present for the audio dataset, all real audio files were segmented into non-overlapping 3-second windows. This was done to increase the number of real samples while also maintaining the integrity of the audio, following techniques commonly used in speech-based deepfake detection (Bird & Lotfi, 2023).

2. **Mel Spectrograms:**
   Each audio segment was then converted into a 128-band Mel spectrogram, capturing the frequency-time distributions which are known to highlight artefacts introduced by artificial/synthetic speech.

3. **Normalisation & Resizing:**
   The spectrograms were normalised and resized to 128x128 pixels in order to ensure they were compatible with CNN layers.

7.3.2 Video/Image Model:

1. **Frame Sampling:**
   Every 15 frames were extracted from each of the .mp4 files in order to reduce redundancy and computational load.

2. **Face Detection via MTCNN**
   Faces detected with a $\geq 95\%$ confidence score were cropped and retained. This removed background noise, reduced input variability, and also ensured high-quality training samples.

3. **Image Normalisation**
   The cropped faces were then resized to 128x128 pixels and normalised prior to being used as MobileNetV2 input.

7.4 Model Development and Training

7.4.1 Audio Model: CNN Architecture

A convolutional Neural Network (CNN) was developed in order to classify Mel spectrograms as either real or fake. The architecture consisted of the following:

- Three convolutional layers (32, 64, 128 filters)
- MaxPooling after each convolution
- Flatten layer
- Dense layer (128 units, ReLU)
- Dropout for regularisation
- Final dense layer with 2 unit softmax for classification

This approach follows well-established practices for detecting deepfake audio (Bird & Lotfi, 2023) and spectrogram classification. The audio model was trained with:

- Categorical cross-entropy loss
- Adam optimiser (Adam, K.D.B.J, 2017)
- 80/20 stratified split
- Early stopping to prevent overfitting

7.4.2 Video/Image Model: MobileNetV2 Transfer Learning

Due to the small dataset size and computational limitations, transfer learning was used here. MobileNetV2 (pretrained on ImageNet) was selected for its efficiency. A two-phase approach was used for this training:

1. **Phase One: Feature Extraction**
    - All MobileNetV2 base layers frozen
    - Added classification head
        - GlobalAveragePooling
        - Dense (128, ReLU)
        - Dropout
        - Softmax output
2. **Phase Two: Fine-Tuning**
    - Final 20 layers unfrozen
    - Low learning rate applied
    - Early stopping used

As the model operates on individual face images extracted from video frames, the same pipeline is inherently suitable for classifying standalone images with facial features. Due to this, the same model is used for both image and video inference.

7.5 Ethics and Reproducibility

All of the datasets used were MIT-licensed and made publicly available. This ensures compliance with ethical research practices. No personally identifiable user data was collected or used throughout the entire research process.

To support reproducibility:

- All preprocessing steps were deterministic
- The full codebase is available via Hugging Face
- The deployed application gives transparent logs, downloadable predictions, and consistent model outputs

The methodology has a clear focus on transparency and repeatability, which are essential for research that involves synthetic media manipulation.

7.6 Methodological Strengths & Limitations

The strengths of this methodological approach include publicly available, well documented datasets and deterministic preprocessing pipelines which supports reproducibility and allows results to be compared with prior deepfake detection research. The lightweight architecture used reflects realistic constraints in many organisational environments, increasing the ecological validity of this study. This approach also allows for more transparency, reproducibility, and accessibility due to the deployable proof-of-concept nature of the framework.

In parallel, there are notable limitations in this approach. Both audio and visual datasets are curated rather than collected from real phishing attacks, limiting external validity and relevance to real-world attacks. The visual model is focused on individual frames, rather than full temporal sequences; therefore, temporal artefacts and cross-frame inconsistencies are not explicitly modelled. Further, the scope of this work is restricted to audio and visual modalities; however, its modular design allows for future work additions of more modalities. Alternative, more complex multimodal or transformer-based architectures were intentionally not considered, as they would be difficult to deploy on CPU only infrastructure and would conflict with the goal of a lightweight, practically deployable prototype.

8 Design Specification

This section shows how the audio and visual detection pipelines developed in Section 3 are integrated into a single, modular framework that can easily be deployed. By specifying the layers, interfaces, and requirements of this architecture, this section helps support RQ3 (integration of the models into a unified app) and provides the basis for evaluating operational suitability in RQ4.

8.1 System Architecture

The system is designed to be a modular, multistage pipeline that processes media uploaded by a user, then routes the media to the relevant deepfake detection model. The architecture is made up of three primary layers:

1. **Input & Interface Layer**
    - A browser-based user interface via Hugging Face
    - Supports uploads of .wav, .mp3, .mp4, and image formats such as .jpeg or .png, etc
    - Provides visual feedback, status messaging, prediction labels, and confidence scoring

2. **Processing & Routing Layer**
   - Media type identification through automation
   - Directs inputs to the appropriate preprocessing pipeline
   - Ensures standardised input format for the deepfake models

3. **Model Interface Layer**
   - CNN audio model
   - MobileNetV2 video/image model
   - Generates predictions and confidence scores
   - Writes the output logs to CSV for auditing, evaluation, and logging

4.2 Functional Requirements

1. **Media Upload Capability**
   The system should be designed to allow users to upload different media types for audio, image, or video for analysis.

2. **Automatic Media Type Detection**
   The system should then identify the media input and route it to the correct processing pipeline

3. **Preprocessing Automation**
   The system should then convert audio to Mel spectrograms, extract frames from video media, and detect faces using MTCNN in visual media, and finally resize/normalise images.

4. **Deepfake Detection**
   The system should then classify each media file as either real or fake and return a confidence score output.

5. **Unified Interface**
   Results should be displayed via a web interface to the user. Results should include predictions, confidence values, and snippets of data the AI is using for detection (For better transparency, e.g. detected faces for video model).

6. **Logging**
   All predictions must be recorded via a CSV file for further transparency, evaluation, and reproducibility. This CSV should then be able to be exported for user interaction.

4.3 Non-Functional Requirements

1. **Usability**

The UI should be both simple and intuitive. It should provide quick and efficient analysis for non-technical users and cybersecurity experts.

2. **Performance**

Processing and inference must be efficient on CPU-based hardware due to deployment constraints and availability concerns. Lightweight architectures such as CNN and MobileNetV2 support this requirement.

3. **Reliability**

The app must be consistent with its predictions and produce repeatable results across identical media inputs.

4. **Security**

Only user-uploaded media is processed. No external data is collected. All inputs are handled within a sandboxed cloud app environment to reduce exposure, and no data is stored upon app session closure.

5. **Extensibility**

The system architecture must allow for additional models, such as text-based or URL-based phishing detection, to be integrated in the future.

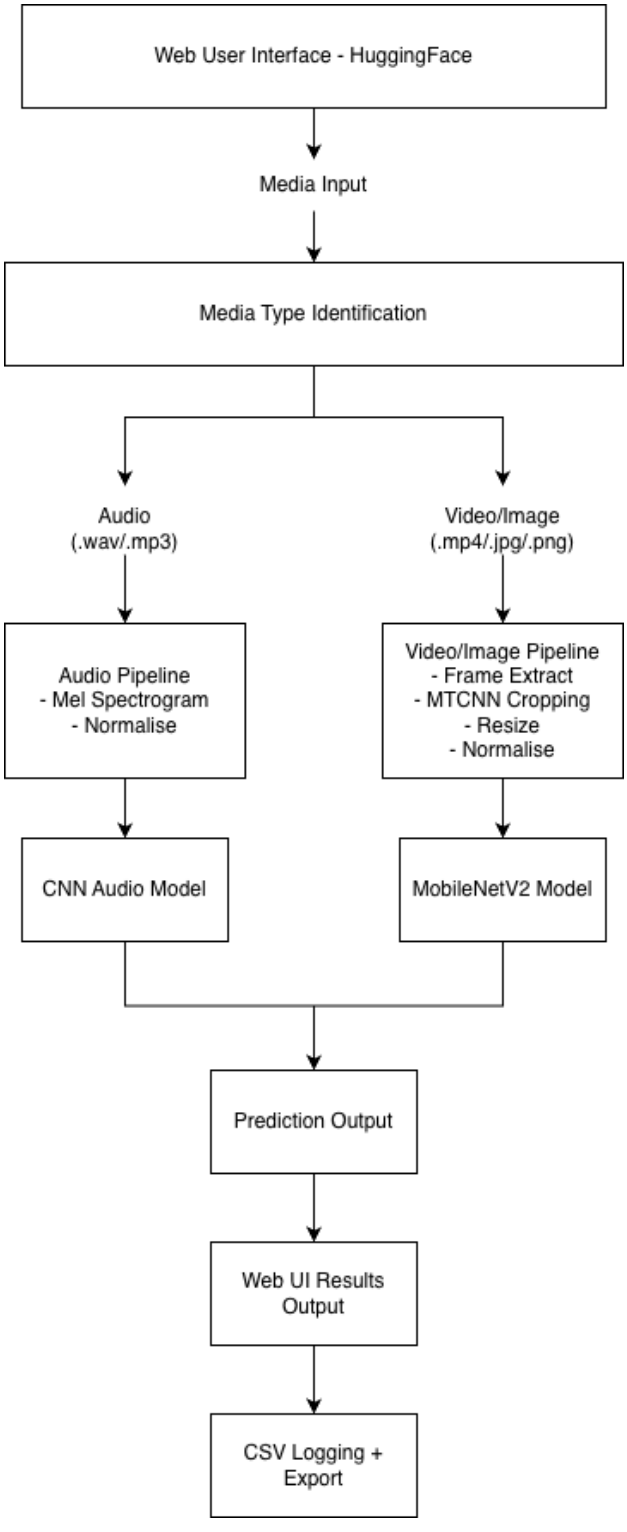


Figure 1

4.4 Design Rationale

Each design decision was made to reflect a balance between:

- Academic validity (e.g., MobileNetV2 for documented performance)
- Practical feasibility (Low computational and hardware cost)
- Reproducibility (Public datasets, deterministic preprocessing, transparent design)
- Usability (Simple upload and check workflow with unified interface)

The decision to use lightweight models was intentional, to focus on CPU-only constraints that are often seen in normal workplace environments, rather than costly GPU performance-based models that are more rarely seen in practical environments. The goal of this design was to develop a functional phishing defence proof-of-concept prototype, rather than a state-of-the-art research architecture/framework.

The system is designed to be intentionally modular so that each model, preprocessing step, and interface layer can be easily redesigned or added upon without needing to redesign the entire pipeline. This supports future expansion for additional modalities such as text, URL or metadata analysis or even more advanced deepfake detection models.

In developing this architecture, several alternative designs were considered. More complex architecture such as EfficientNet, 3D CNNs, GPU dependent deployments and tightly coupled multimodal fusion layers, could potentially create higher benchmark accuracy, however, these options were not used for this research due to the significant increase in computational cost. Additionally, they would further complicate deployment on typical CPU-only infrastructures and reduce overall transparency by intentionally comparing the framework to lightweight audio and visual models within a simplistic and accessible UI. In doing so, the design remains aligned with the research focus on a practical and deployable proof of concept for phishing defence workflows, rather than a conceptual, state-of-the-art benchmark model. These architectures also do not align with the research questions set out in Section 1 (RQ1-RQ3).

9 Implementation

This section shows how the framework design outlined in Section 4 is realised as a working prototype. In doing so, it operationalises the integration-focused aspects of RQ3 by showing how the two modalities are merged into a single tool and prepares the ground for RQ4 by exposing the practical constraints, engineering trade-offs, and deployment decisions that are evaluated later.

9.1 Audio Detection Model

All audio files were pre-processed using the same pipeline as described in Section 3: Raw clips are segmented into 3 seconds windows, converted to 128x128 Mel Spectrograms and normalised. At runtime, the pretrained CNN (3 convolution-pooling blocks followed by a dense layer and SoftMax output) is loaded, the uploaded file is passed through this pipeline, and a REAL/FAKE label with a confidence score is returned to the interface. All further architectural and training details for this model are covered in Section 3.4.1.

9.2 Video/Image Detection Model

For image and video inputs, the deployed system reuses the preprocessing pipeline described in Section 3: every 15th frame is sampled from the uploaded file, MTCNN extracts face crops with ≥95% confidence, and each face crop is resized to 128x128 and then normalised before being passed over to the MobileNetV2 classifier.

At inference time, the model predicts a REAL/FAKE label and confidence score for each detected face. For video inputs with numerous detected faces per video frame, the per face predictions with the highest confidence are then aggregated to produce a single decision for the video. Due to the classifier being trained exclusively on face crops, the same pipeline is also used for still images that contain faces, allowing a single visual model to handle both static and dynamic media.

9.3 Unified Multimodal Application

The two models are exposed through a Gradio Blocks interfaces deployed on Hugging Face Spaces and organised into three tabs for audio, image and video. On app startup, the app authenticates with the Hugging Face Hub, downloads the latest .keras artefacts from private repositories and loads them into memory.

Each tab applies the corresponding pipeline (audio → Mel spectrogram, image/video → face detection and cropping) before then calling the relevant classifier and displaying the REAL/FAKE label, confidence score and previews of relevant artefacts being analyzed (e.g. mel spectrogram or sample face crops) in the browser.

9.4 Deployment

The application is deployed via Hugging Face Spaces, offering a containerised environment for running Gradio application with reproducible dependencies. The deployment includes:

1. A Git repository containing app.py, training notebooks via .ipynb for model training transparency, dataset archive for testing and transparency, and a requirements.txt file for app requirements.
2. A dependency specification including TensorFlow 2.19.0, Gradio 4.x, Librosa 0.10.1, OpenCV-headless, MTCNN, Hugging Face Hub client, and related libraries.
3. A secure environment variable (HF_TOKEN) used for authenticated downloads of private model artefacts to prevent model tampering.

After the application launches, the platform builds the environment based on a requirements.txt file. It then executes app.py and loads the UI on a public URL. The system then retrieves the latest .keras model files from the private repositories, ensuring consistency between development and deployment.
The deployment shows how a multimodal deepfake detection system can operate entirely within a lightweight, browser accessible cloud service without costly dedicated GPU hardware. This reinforces the feasibility of integrating similar tools into real world phishing defence workflows.

To summarize, the implementation resulted in a fully functional prototype that enables the trained models through a unified web UI, with consistent preprocessing, model loading and result logging pipelines. The intentional usage of CPU-only infrastructure, lightweight dependencies, and ephemeral file handling reflects the constraints under which many security teams would realistically deploy such a tool. These implementation decisions not only satisfy the integration objective outlined by RQ3 but also create concrete behaviours and performance characteristics that can be evaluated in section 6, which addresses RQ1, RQ2, and RQ4.

10 Evaluation

The purpose of this section is to provide an empirical evaluation of the proposed framework in order to address RQ1, RQ2, and RQ4.

10.1 Evaluation Setup

Both of the models were trained and tested using stratified splits in order to preserve class balance. Evaluation was performed only on withheld test sets to ensure that the models weren't exposed to any samples during training. Model performance was assessed via:

- Accuracy
- Precision, Recall, F1 score
- Confusion matrices
- Training/validation loss and accuracy curves
- Manual inspection of representative samples

Notably for the image/video model, evaluation was conducted at the face-crop level, consistent with the MTCNN preprocessing pipeline.

10.2 Audio Model

10.2.1 Dataset & Experimental Setup

The audio classifier was evaluated on 1989 test samples. Each sample represented a three-second Mel spectrogram segment. The dataset included 1739 fake samples and 250 real samples. This reflected a large imbalance that favours fake samples over real samples. Each sample was processed into a 128x128 Mel spectrogram using the preprocessing pipeline described in Section 3.

10.2.2 Quantitative Performance

Table 1 represents the audio Model's classification results on the held out test set:

Table 1

Class	Precision	Recall	F1 Score	Support
REAL	0.97	0.80	0.88	250
FAKE	0.97	1.00	0.98	1739
Accuracy	-	-	0.97	1989
Macro Average	0.97	0.90	0.93	1989
Weighted Average	0.97	0.97	0.97	1989

The model achieved a 97% accuracy rating, demonstrating a strong generalisation on unseen samples. While the model creates more false positives than false negatives (low recall), this behaviour is still acceptable in a phishing context, since missing a deepfake (a false negative) creates greater risk than incorrectly flagging a real sample.

Figure 2 displays the confusion matrix for the audio classifier.

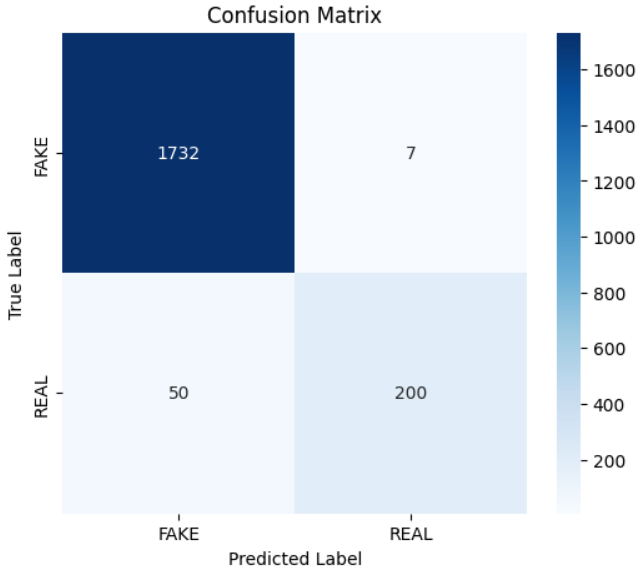


Figure 2

Most errors occur where the REAL audio is misclassified as FAKE, which is consistent with the lower recall for REAL speech compared to FAKE speech. As REAL samples are in the minority class, the model has fewer examples to learn their characteristics, leading to most errors occurring where REAL audio is misclassified as fake. However, the classifier demonstrates strong robustness to synthetic samples and a reliable generalisation across unseen voices.

Training and validation curves revealed stable convergence with closely tracking accuracy and loss, indicating no significant overfitting. *Figure 3* demonstrates the accuracy curve while *Figure 4* demonstrates the loss curve.

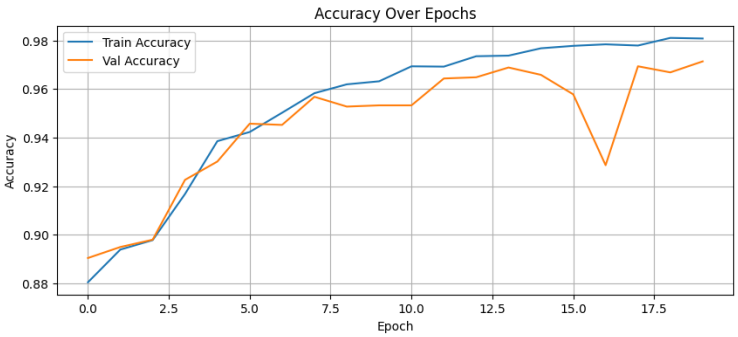


Figure 3

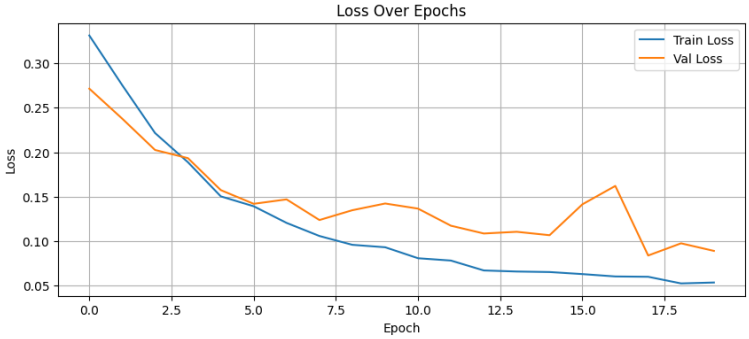


Figure 4

10.3 Image/Video Model Evaluation

10.3.1 Dataset & Experimental Setup

The visual classifier was evaluated with 235 face crops being extracted from the FaceForensics++ test videos, which comprised of 124 REAL samples and 111 FAKE samples. Inputs were resized to 128x128 RGB and then passed through the MobileNetV2 classifier.

10.3.2 Quantitative Performance

Table 2 shows the visual classifiers performance:

Table 2

Class	**Precision**	**Recall**	**F1 Score**	**Support**
REAL	0.81	0.95	0.87	124
FAKE	0.93	0.75	0.83	111
Accuracy	-	-	0.86	235
Macro Average	0.87	0.85	0.85	235
Weighted Average	0.87	0.86	0.85	235

In comparison to the audio model, the accuracy is lower at 86%. The FAKE precision is high (0.93), rarely labelling real faces as fake. The FAKE recall is lower (0.75). This indicates that some fake faces are misclassified as real.

Figure 5 shows the confusion matrix for the visual classifier. The error trend shows the model is more conservative about classifying visual media as FAKE. It appears to a trend of classifying ambiguous faces as real. This aligns with the FAKE recall of 0.75.

Training and validation curves stabilised at similar accuracy levels and steadily decreasing loss, again suggesting successful transfer learning without significant overfitting.

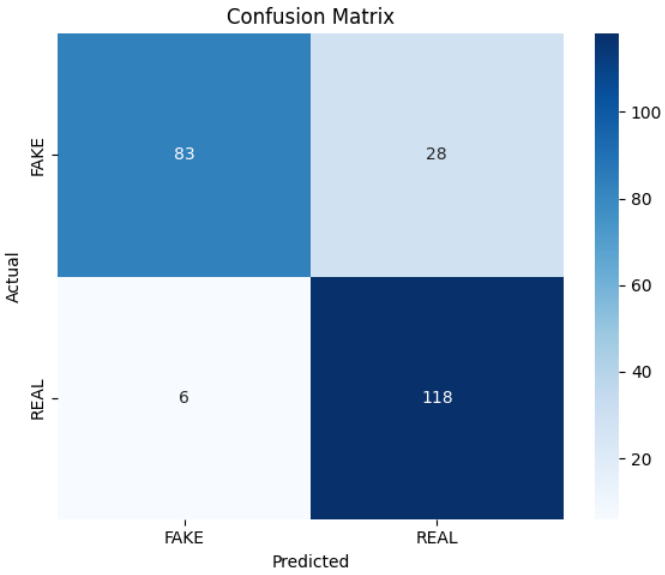


Figure 5

10.4 Multimodal Application Evaluation

The application inference at interactive speeds is appropriate for analyst facing phishing defence workflows with typical speeds for audio < 1-10s, for images < 1-10s and for videos (dependent on length) <10-30s. No stability issues or major failure cases were observed during application evaluation.

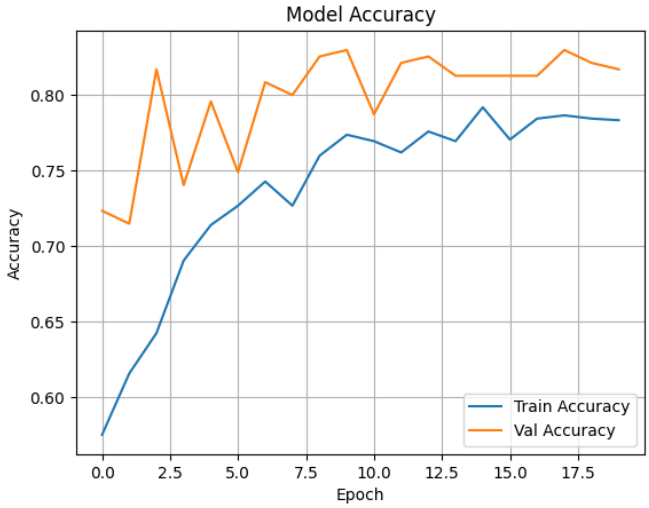


Figure 6

10.5 Cross Model Comparison & Summary of Findings

Table 3

Metric	Audio Model	Image/Video Model
Accuracy	0.97	0.86
Macro F1	0.93	0.85
FAKE Precision	0.97	0.93
FAKE Recall	1.00	0.75
REAL Precision	0.97	0.81
REAL Recall	0.80	0.95

The audio model has an overall accuracy rating of 97%. Along with a recall score of 1.00 for fake samples, making it highly successful in detection. In contrast, the visual model achieves an accuracy of 86% and high precision, but with a low recall score on fake samples. This demonstrates with the challenges involved in detecting subtle facial manipulations. Training curves for both models shows stable convergence and no significant overfitting. The deployed multimodal application replicates notebook observations accurately and reinforces the feasibility of using a lightweight, CPU friendly approach in model development for practical phishing defence use cases.

10.6 Discussion

From a methodological standpoint, application is multimodal in input, not in decision making. Both models are evaluated independently, and no fusion mechanism combines their outputs together. The evaluation also does not include robustness tests against common real-world distortions such as heavy compression, background noise, adversarial artefacts, etc. However, despite these limitations, the deployed application reproduces notebook results consistently and demonstrates that a lightweight, CPU friendly, unified deepfake detection framework can support an interactive phishing defence workflow. Future work should focus on the expansion of dataset diversity, incorporating temporal modelling for video, introducing multimodal fusion and evaluating robustness against realistic attack conditions.

From an academic point of view, these findings broadly support prior work on deepfake detection while also highlighting the trade offs introduced by lightweight, CPU oriented design. The strong performance from the audio model is consistent with Bird and Lotfi's (2023) observation that CNNs that are trained on Mel spectrograms can reliably detect AI generated speech, even on relatively small datasets. The visual models 86% accuracy tends to miss some fake faces while rarely mislabelling real faces. This reflects the challenges seen in FaceForensics++ based studies such as Rössler et al., (2019) , where subtle facial alterations and compression artefacts make frame level detection challenging. Compared with the more complex multimodal literature (e.g. Salvi et al., 2023; Wang et al., 2025; Bekheet et al., 2024), the results here are modest in absolute terms but also demonstrate that reasonable strong performance can be achievable when using compact models that are deployed over CPU only infrastructure. Based on these findings, the evaluation collective answers RQ1 and RQ2 by quantifying the benefits and limitations of the chosen audio and visual classifiers and contributes to RQ4 by showing how design decisions motivated by real world like constraints influence achievable accuracy, robustness and error patterns.

11 Conclusion and Future Work

Research Question Answers:

1. **RQ1:** A CNN trained on Mel spectrograms is able to detect artificial speech with high performance under hardware constraints. Displaying this by achieving 97% accuracy and 1.00 recall for fake samples. This indicates strong suitability for flagging voice-based phishing attacks.
2. **RQ2:** A MobileNetV2 classifier trained on extracted face crops from FaceForensics++ reaches 86% accuracy with high precision but lower recall on fake faces. This indicates that single frame visual detection remains challenging and more likely to miss subtle visual manipulations.

3. **RQ3:** Both audio and visual models can be integrated successfully into a unified, browser-based application that outputs real time predictions, confidence scores, CSV logging at interactive speeds on CPU only infrastructure, showing the practical feasibility of the proposed multi-modal deepfake detection framework.

4. **RQ4:** The evaluation and deployed application tests highlighted key limitations and failure modes, such as sensitivity to dataset bias, lack of temporal modelling and absence of complex architecture like multimodal fusion. However, the findings also show that a lightweight, transparent, multi-modal prototype can realistically support analyst facing phishing defence workflows with accurate results.

The deployment of the application is a key outcome of this research. In contrast to many other studies that only report on offline, high end, state-of-the-art metrics, this project delivers a functioning proof of concept tool that can log predictions, exposes model outputs in a transparent manner and operates at interactive speeds on CPU only infrastructure. Overall, the research demonstrates that a lightweight, multimodal deepfake detection framework can be practically integrated into a phishing defence workflow, while also highlighting clear directions for the improvement on real life effectiveness, generalisability and multimodal reasoning.

Prior work explored in related literature shows that such fusion can significantly improve robustness, especially when a modality is noisy or partially unavailable (Salvi et al., 2023; Bekheet et al., 2024). Another avenue future work should focus on is the improvement of the video model through the inclusion of 3D CNNs, recurrent networks or transformer-based to detect high quality deepfakes that use more advanced deepfake technology for hyper realistic visual mimicry. Similarly, future work should include more diverse and challenging datasets for both models.

The proof-of-concept prototype developed in this research is mostly relevant to security operations centre (SOC) teams, fraud and risk units, threat detection and research teams, and incident responders who have regular exposure to phishing attacks. For these relevant stakeholders, a lightweight, browser-based tool can offer a quick solution to flagging potentially manipulated media without requiring GPU class hardware and offers a pragmatic way to augment current phishing response workflows. Email gateway vendors, managed security service providers and cloud security platforms could integrate this framework or similar functionality into their own security architecture as an additional inspection stage for detections, investigation or response. It could be used for active phone/video calls, audio, image or video file attachments/downloads, and links that utilise media for social engineering. Moving towards commercial deployment would require further training, hardening, extensive testing on real phishing attack data, privacy and compliance reviews and clear communication of model limitations to avoid over reliance on automated decision making.

References

Adam, K.D.B.J., 2017. A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 1412(6).

Alshingiti, Z., Alaqel, R., Al-Muhtadi, J., Haq, Q.E.U., Saleem, K. and Faheem, M.H., 2023. A deep learning-based phishing detection system using CNN, LSTM, and LSTM-CNN. Electronics, 12(1), p.232.

Aslam, S., Aslam, H., Manzoor, A., Chen, H. and Rasool, A., 2024. AntiPhishStack: LSTM-based stacked generalization model for optimized phishing URL detection. Symmetry, 16(2), p.248.

Bekheet, A.A., Khoriba, G. and Ghoneim, A.S., 2024. Development of a multimodal framework for deepfake detection: combining visual and audio analysis. In Proceedings of the 10th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS 2024).

Bird, J.J. and Lotfi, A., 2023. Real-time detection of ai-generated speech for deepfake voice conversion. arXiv preprint arXiv:2308.12734.

birdy654 (n.d.) DEEP-VOICE: DeepFake Voice Recognition [Dataset]. Kaggle. Available at: https://www.kaggle.com/datasets/birdy654/deep-voice-deepfake-voice-recognition (Accessed: September 10th 2025).

Catal, C., Giray, G., Tekinerdogan, B., Kumar, S. and Shukla, S., 2022. Applications of deep learning for phishing detection: a systematic literature review. Knowledge and Information Systems, 64(6), pp.1457-1500.

Ejaz, A., Mian, A.N. and Manzoor, S., 2023. Life-long phishing attack detection using continual learning. Scientific reports, 13(1), p.11488.

Ige, T., Kiekintveld, C. and Piplai, A., 2024. Deep learning-based speech and vision synthesis to improve phishing attack detection through a multi-layer adaptive framework. *arXiv preprint arXiv:2402.17249*.

Jha, A., Bhat, B., Abhay, C.M., Hegde, C. and Srinath, R., 2025, July. Real-time audio-visual deepfake detection using multimodel pipeline in video conference. In Tenth International Conference on Multimedia and Image Processing (ICMIP 2025) (Vol. 13701, pp. 146-156). SPIE.

Mehdi Gholampour, P. and Verma, R.M., 2023, April. Adversarial robustness of phishing email detection models. In Proceedings of the 9th ACM international workshop on security and privacy analytics (pp. 67-76).

Pei, G., Zhang, J., Hu, M., Zhang, Z., Wang, C., Wu, Y., Zhai, G., Yang, J., Shen, C. and Tao, D., 2024. Deepfake generation and detection: A benchmark and survey. arXiv preprint arXiv:2403.17881.

Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J. and Nießner, M., 2019. Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1-11).

Salvi, D., Liu, H., Mandelli, S., Bestagini, P., Zhou, W., Zhang, W. and Tubaro, S., 2023. A robust approach to multimodal deepfake detection. Journal of Imaging, 9(6), p.122.

Tamal, M.A., Islam, M.K., Bhuiyan, T., Sattar, A. and Prince, N.U., 2024. Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning. Frontiers in Computer Science, 6, p.1428013.

Tan, D., Yang, Y., Niu, C., Li, S., Yang, D. and Tan, B., 2025. A review of deep learning based multimodal forgery detection for video and audio. Discover Applied Sciences, 7(9), p.987.

Wang, L., Zhao, J., Zhang, X., Guo, X., Yuan, Y., Zhang, T., Chu, J., Jiang, Y., Yang, X., Jin, L. and Zhang, C., 2025. ERF-BA-TFD+: a multimodal model for audio-visual deepfake detection. Vicinagearth, 2(1), pp.1-12.