Enhancing Deepfake Detection with Multi-Modal Transformers

MSc Research Project
Data Analytics

Sahana Hombal
X23207655

School of Computing
National College of Ireland

Supervisor: Teerath Kumar Menghwar

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: … Sahana Hombal ……………………………………………………………………………

Student ID: ………x23207655…………………………………………………………………………….……

Programme: ………Data Analytics……………………… **Year:** ………2024…………..

Module: …MSc Research Project……………………………………………………………….………

Supervisor: …………Teerath Kumar Menghwar…………………………………….………

Submission Due Date: ……………29/01/2025………………………………………………………………….………

Project Title: …Enhancing Deepfake Detection with Multi-Modal Transformers……

Word Count: ………7823………………… **Page Count**……20……………………………………..

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: ……………Sahana Hombal………………………………………………………………

Date: ………………29/01/2025…………………………………………………………………

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):	

Enhancing Deepfake Detection with Multi-Modal Transformers

Sahana Hombal
23207655

Abstract

The recent development of deepfake technology has made it more challenging to verify the accuracy of digital content, especially when it comes to preventing manipulation and protecting cybersecurity. Advanced multi-modal false information is difficult to identify as current detection methods tend to utilize single modalities, such as visual or audio data only. These limitations are however addressed by this research through proposing a multi-modal deepfake detection system that adopts transformer architectures in analyzing both image and audio data. The proposed system is designed to improve the accuracy and efficiency of the detection techniques to afford a complete solution for detecting manipulations of media content. For the image-based detection, the study used Deepfake and Real Images Dataset and for the audio-based detection, the study used the Fake-or-Real Dataset. Real and fake images were predicted using ResNet50, VGG16, MobileNetV2, and InceptionV3 with adjusted layers; a convolutional and recurrent model was designed to perform on the audio data. Data enhancement techniques, normalization and spectrogram formation for audio corpus used for training and testing are applied for better accuracy. Performance of the models was measured based on parameters such as accuracy, precision, recall, and F1-score in order to make the assessment modality exhaustive. The data confirm the efficiency of the multi-modal system with a ResNet50 model accuracy of 94.5% for image detection, and the CNN-LSTM 91.4% F1-score for audio detection. By combining the spatial and temporal elements of the system, these results show how the method excels at identifying minute artifacts across media. However, the system has using difficulties when considering distortions and conditions for real markets. This work sets a new state-of-the-art in terms of deepfake detection, providing important implications for media authentication, cybersecurity, and countering fake news. As for the future work, the main areas to improve the system is to make it work better in real environment and to integrate more modalities to improve the strength of the detection system.

Keywords: Deepfake Detection, Multi-Modal Transformer, PyTorch, TensorFlow, Media Forensics, CNN-LSTM Hybrid Model, Digital Content Authenticity

1. Introduction

Deepfake was introduced by using artificial intelligence and deep learning technologies, which brought extreme changes with itself to multimedia content generation. These technologies apply deep learning frameworks to create convincing artificial media in the form of fake audio, images, and videos that can easily be mistaken for the real thing. This is the basis for the potential and risks of deepfakes, which could include Entertainment, accessibility and education This ability to effectively control the media raises serious questions of identification, misinformation, and security risks. Since it is possible to use developing technology and produce very real fake media, it has been used for a number of negative purposes, such as distributing false information, stealing money during fraud, and identity theft scams. For example, deepfake videos have been used in assigning statements of politicians that they did not make while artificially generated audio is used in the voice scam attacks that poses a threat to lives and corporations.

This is why the detection of deepfake content is so problematic because its quality is so close to real images and real audio. Traditional methods of identification, mainly based on the simplest image or voice recognition procedures may no longer work well enough due to the increased use of more advanced deepfake generation methods such as GANs. Most of these tools have the ability and capability of generating nearly accurate images and videos that show facial movements, voice match to the lip movements. Even worse, generative models that have recently emerged such as Google WaveNet, Deep Voice, Tacotron, and those based on GANs make detection even more challenging because of the near perfect copying of human behaviors that they create. Deepfake photographs, for example figure 1, can contain pixel level changes that are difficult to detect by frequently used algorithms, and if deepfake is applied to both the audio and the related video, the result is multi-modal fraud that is highly effective and believable.

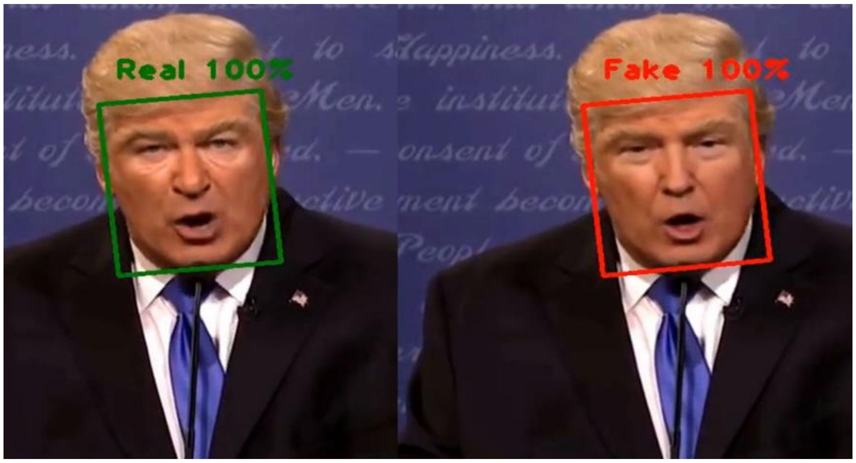

Figure 1. Example image of original and deepfake

These developments are going to raise major ethical issues that go beyond technological progress. Deepfake breaks the concepts of privacy and consent since a person cannot decide whether their voice or picture is impersonated. Figure 1 is best example. (Anon., n.d.)Second, the need for the detection system becomes much more critical given its capacity to produce false information and damage reputations. This means that in addition to preventative measures like regulation, it is crucial to invest in and develop independent, accurate, and efficient detection tools.

This project was created in order to address these issues by developing a better system that is unique in its ability to identify deepfake audio and images (J. Asan, 2023). In order to achieve this, the project aims to use deep learning techniques and algorithms, such as CNNs and multi-modality, to take picture and audio spectrogram properties into account. The goal is to find patterns like a frequency shift in the audio track or tiny grains in the videos pixels that should not be present but are frequently missed in deepfakes.

In the audio domain, a model uses Mel spectrograms, which also contain synthetic audio patterns, to recognize manufactured speech. CNN analyses facial geometry and textures in photographs to look for characteristics that are likely the result of deepfake creation (Chen, 2024). The developed method aims to bridge the gap between multi-modal detection, which involves examining the correlation between the audio and video components of the deeper bogus content, and purely single-modal detection, which involves either audio or image analysis.

In this work, multiple pre-trained deep learning models such as ResNet50, VGG16, MobileNetV2, and InceptionV3 along with custom built CNN models are used for the detection of deepfakes and have been compared. It is planned to enhance these models through the fine-tuning of the models and other hyperparameters in order to increase the model's accuracy and its computations. The aim of this project is to identify secondary artefacts that distinguish the deepfake from a real person, such as unnatural lighting and structure variations, which increase the method's chance of scalability and practicality when it comes to images and videos, instead of identifying the information that was changed during the creation of the deepfake itself (Qi, 2021).

2. Related Work

2.1 Deep Learning for Deepfake Detection

The implementation of deep learning to identify deepfakes has grown in popularity, especially as deep learning can analyze images and videos in the context of complex patterns. One of the most popular techniques is transfer learning; examples of this include ResNet, VGG, and Inception, which are optimized using large data sets, such as ImageNet. These are deepfake detection models that have already been trained in order to add extra layers for the binary labeling of real and fake content. For example, the examination of the FaceForensics++ dataset shown that deep learning models like Xception and ResNet are ideal for identifying common manipulations like video reuse and face swapping (Rossler, 2019).

The creation of custom CNN architectures has been studied in connection with pre-trained networks. The designs mentioned above are specifically designed to detect artifacts resulting from deepfake generating processes. Basically, they like to focus on differences in elements like lighting, texture, or facial emotions that happen during it (Dolhansky, 2020). For example, if trained from scratch, customized CNN architectures have been used to secure high detection rates as indicated by other effective feature extraction techniques. The second stream of deepfake detection targets generative models known as Generative Adversarial Networks (GANs). To identify such content, these techniques use traces left on the materials used to create modified media. Small elements like lighting or an image's texture are frequently examined to identify abnormalities, usually in terms of red-lighting to recreate using deepfake. In the case of video-based detection, results have shown that architectures comprising of both CNNs and RNNs give the best results. As a result, these models may identify differences in both frame-by-frame and object movements inside images and audio sequences.

2.2 Developing Method

Although there has been a lot of progress in recent years, there are still certain challenges in detecting deepfakes. The biggest challenge is related to the fact that deepfakes are becoming ever more advanced and have high-quality images and audios. Modern generative methods such as GANs lead to the generation of content that has a realistic visual representation with minimal traceability (Karras, 2020.). Additionally, there are a lotmore actual samples than fake ones in deepfake data because to the imbalanced quantity of images and videos.

This results in bias sensitivity to false positives and false negatives in the new model (Johnson, 2019). Another issue is computational complexity, as complicated models such as ResNet50 are only suitable for contexts with limited processing power and need an important number of resources to train and use.

This research uses pre-trained models, including ResNet50, VGG16, MobileNetV2, and InceptionV3, to address these problems through transfer learning (Simonyan, (2014).). The outstanding extraction capabilities of these models make them ideal for detecting deepfakes. Additionally, trained CNN models are built and compared in order to understand the variation between standardized models and models that have been tuned for certain applications. Furthermore, hybrid approaches are thought to enhance the effectiveness of both time and space analysis as well as image detection (Wang, (2021).).

However, there are still several limitations in the deepfake detection research: while Current models, especially those based on modern GAN technology, are unable to identify small manipulations produced by emerging deepfake generating technologies. Current datasets are also not diverse enough in terms of types of manipulations, therefore the detection models are not able to learn many types of scenarios. In addition, the present models demand for low memory usage or real-time performance remain to be challenging, demanding scalable solutions.

3. Methodology

This section outlines the process of collecting, pre-processing, transforming, modelling, and evaluation of the data for detecting deepfakes. The methodology makes sure a systematic approach to achieve Accurate results in both visual and audio classification tasks.

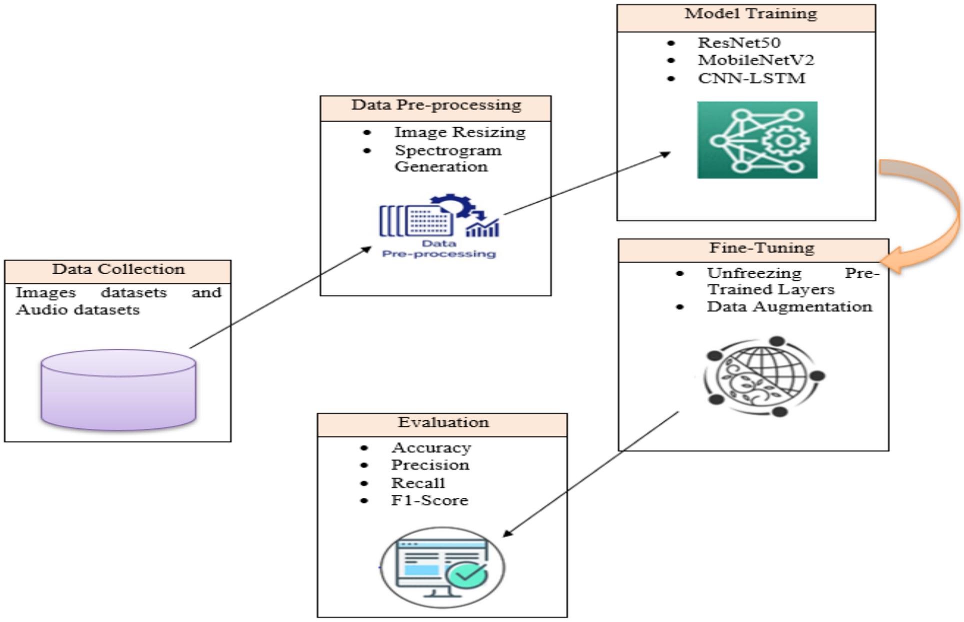

Figure 2: Workflow of the deepfake detection system, showing the progression from data collection to evaluation

3.1 Data Collection

The datasets for this project are the Deepfake and Real Images Dataset and the Fake or Real (FoR) dataset both of which are available on Kaggle. The Deepfake and Real Images Dataset focuses on visual data, containing labeled images of two categories: Fake or artificially Generated depicting a set of images that have been altered or deep faked and Real which shows the raw form of images (Bengio, 2013). This dataset has been collected for the purpose of training new image classifiers to detect fake content on the Web.

The Fake-or-Real (FoR) Dataset includes various research and fake speech samples. It gathers information from actual human readings from several sources, such as the Vox Forge Dataset and the Arctic Dataset, as well as benchmark TTS systems like Google Wave Net and Deep Voice 3. The dataset is available in multiple versions: the first one is the 'for-original', the second the 'for-norm', the third is short two second samples called 'for-2sec,' and the last one is the fake environment imitating 'for-re rec'. A wide range of data types and sources guarantee effective training for speech-based classifiers.

Both datasets were organized into three subsets: The training set containing the data used to feed the models; the validation set containing data used to fine-tune parameters of, and test models in the middle of training; and the test set that contains data to test the performance of the models on unknown data they were never trained on.

3.2 Data Cleaning and Pre-Processing

Further, before using the datasets, a number of operations to clean the data were conducted to represent interaction with the Deep learning models. A few of the files were removed because they were either corrupted or missing, ensure that only clean data was used. Additionally, the accuracy of the labels was verified, and balancing techniques were applied to equalize the number of real and fake samples in the datasets (Chen, 2024).

With regards to image data, preprocessing included the resizing of all images to the format 224 by 224 pixels as were expected by the pre-trained models used within this project. To do this the pixel values were scaled to a range of 0 to 1 so that the values did not fluctuate between large values during training. The augmentation of the training dataset was performed in order to minimize the overfitting and improve generalization. These types of changes added more variation such as rotations, shift, zoom and horizontal flip that closely resembles the variations in the image data (Karras, 2020.).

In the data preprocessing of the audio data, the raw audio signals were transformed into spectrograms because of simple in analysis in the frequency domain. Truncation was used here for audio clip normalization and especially for models trained on the context of for-2sec version of this corresponding dataset. As for audio data, all extracted audio signals had their amplitudes normalized for compatibility of equipment and signal levels. In addition, signal balancing was performed for gender differences in synthetic and, real speech samples provided in the for-norm version.

3.3 Data Visualization and Transformation

In order to prepare the datasets for the next deep learning models, a number of data cleaning and pre-processing techniques were performed. Lost or damaged data files were identified and deleted to prevent contamination and to ensure high data quality. Also, the accuracy of the labels was checked by the authors, and balance was applied to Fake and or Real data sets in the data sets. In case of image data preprocessing included in the proposed solution, resizing of images to a fixed dimension of 224×224 pixels were performed to fit the input of the used pre-trained models in the project (Ojala, n.d.).

At pixel space were scaled to the range [0, 1] in order to make the training process more stable. Preprocessing procedures were also used on the training set to show additional related patterns for better generalization of the model rather than overfitting (Mugoya & Kampfe, 2010). These transformations were the rotations, shifts, zooming, horizontal flipping which are the realistic variations in image data.

For the audio data, preprocessing A raw audio signal was transformed into a spectrogram for frequency domain analysis. Questions and answers remained distinct in all of the models, while truncation was used to normalize the length of audio clips, especially for the models trained with the for-2sec version of the dataset. Further equalization was applied to counter the percentages of female and male samples in synthetic and real speech samples as in for-norm version of the dataset.

3.4 Modelling and Experimentation

It used several machine learning algorithms such as transfer models and one newly trained convolutional neural network (CNN) model. In Fine-tuning: The first step in our research proposal is the fine-tuning of pre-trained deep models ResNet50, VGG16, MobileNetV2, and InceptionV3 on the Fake/Real samples. These models used the reservoir of information obtained from huge data bases such as ImageNet to obtain feature detectors that were highly powerful (Sandler, 2018). In order to fit these models for this particular task, the final layer of each pre-trained models was substituted with GAP layer, followed by a single neuron sigmoid layer for binary classification. First the pretrained layers hold onto the pretrained weights and only the new layers were allowed to be trained. Following that, in fine-tuning, layers of the pre-trained models, not all of them, were ‘unfrozen’.

A custom CNN was also created to evaluate the performance difference of the model being proposed when compared with the pre-trained models (Huang, 2020). This network consisted of multiple convolutional layers with an ever-increasing filter size to capture the hierarchical decomposition, and max-pooling layers that decrease spatial dimensionality while preserving as much of the important features as possible. The flattened output was then being fed into a fully connected dense layer, and to minimize the problem of overfitting a dropout layer was used. The last layer of output employed sigmoid activation to label the input either as Fake or Real.

The experiments were carried out with different combinations of the model’s configurations and parameters. For the initial training phase, learning rate was fixed at 1e-4 while for the fine-tuning process, learning rate is 1e-5. Batch size of 32 was used, and dropout was taken to be 50% for reducing the overfitting problem. The intercept took place to stop the trainings when the validity loss remained constant, and the saver only stored the models with high validity accuracy.

3.5 Evaluation

In order to compare trained models, several metrics were used to evaluate models on the test set. In this case, accuracy, the percentage of samples for each target variable that are properly classified, was most frequently applied. To estimate the performance of the model on the task of Identifying between Fake and Real images, precision and recall rates have been used. The F1-score that is a combination of precision and recall, offered an overall measure of the model’s performance (Beloglazov & Buyya, 2015). A confusion matrix table was developed in order to provide a more detailed view of the resources into true positive, true negative, false positive, and false negative categories.

A function of epochs was used to describe statistical information, such as accuracy and loss, of the training and validation sets in order to evaluate the training process. The resulting visualizations displayed which model overfitted the data and how the models are resolved. Additionally, a comparison of the initialized models and the custom CNN was done with regard to its accuracy and computational complexity. While the accuracy of the previous levels was better, the fine-tuned CNN provided information about task-specific compared to general-purpose transfer learning architectures.

4 Design Specification

This section describes the design properties of the deepfake detection system including the techniques, architectures, frameworks, and related specifications. The focus of the design lies in the development of a sound and feasible approach to identify deepfakes in image and audio space.

4. 1. Techniques and Methodologies

A CNN-LSTM model, transfer learning, and custom CNNs are used in the deepfake detection system to solve the difficulties in images and audio-based deepfake identification.

4. 1.1 Techniques for Image Data

The pre-trained models such as ResNet50, VGG16, Mobile Net V2, and INCEPTION V3, the transfer learning approach is taken into consideration for image-based deepfake detection. These models require little training data because they make use of the data that an algorithm learns from huge data sets like ImageNet. By fine-tuning certain layers of the models, the last ones are able to focus only on the aspects which are characteristic to deepfake artifacts. A specific convolutional neural network (CNN) is also created for experimentally testing task-optimized designs where it is expected that unusual lights, insignificant edges, or even modified textures will be separated from pre-trained models.

4.1.2 Techniques for Audio Data

In the case of audio-based deepfake detection, the CNN-LSTM is applied. The CNN layers extract the features from the spectrograms derived from the audio samples, on the other hand, LSTM layers captures the temporal behavior or dependencies of the audio signals. This approach is capable of addressing properties of the frequency domain that are spatial as well as temporal features that are critical to detecting synthetic speech artifacts.

4.2. System Architecture

The architecture for the deepfake detection system is modular, designed to handle both image and audio data streams. It comprises the following components.

4.2.1 Preprocessing Pipeline

The conversion of the initial input data into a format that the models can use for training or prediction is supported by preliminary processing. For location data it involves the resizing of all location photos to a standard of 224 x 224 pixels, normalization of all location photos to a range of 0 or 1 and various data augmentation techniques to increase the diversity of the data set (Chen, 2024). For audio data, spectrograms are generated from basic waveforms, and the Librosa package is used to extract features such as the Mel-frequency cepstral coefficients.

4.2.2 Model Framework

TensorFlow and Keras are used to implement the models, using their powerful APIs for deep learning model construction and training. The system includes:

- Transfer learning models: include ResNet50, VGG16, MobileNetV2, and InceptionV3 are used for image data.
- Custom CNN: used to identify deepfake image artefacts.

- Hybrid CNN-LSTM: This method combines temporal and spatial analysis to detect deepfakes in audio.

4.2.3 Training and Evaluation Workflow

Supervised learning is used to train the models on labelled datasets. Model checkpointing and early stopping provide high performance without overfitting. To evaluate the performance of the model, evaluation measures like as accuracy, precision, recall, and F1-score are computed.

4.3 Framework and Requirements

4.3.1 Framework

The deepfake detection system can be effective, there are clear assumptions and expectations that must be met in order to best utilize it's set up. The development of the deep learning system used this TensorFlow and Keras environment since it has many tools to create and fine-tune the DL system. For signal preprocessing to audio data and feature extraction, a librosa is used and the extracted features include Mel-frequency cepstral coefficients (MFCCs) (McFee, n.d.). Data distributions, spectrograms and the results of model training are visualized using Matplotlib and Seaborn.

4.3.2 Computational Requirements

The usage of a GPU environment essential for training and assessing deep learning models, specifically transfer learning and hybrid configurations. The suggested GPU is above 8 GB (Nvidia RTX 2070 or better), 16 GB of RAM and 50 GB for datasets and checkpoints. Another requirement is that the stack must contain TensorFlow 2.x, Python 3.7 and later, in addition to other libraries like SciPy and NumPy.

4.3.3 Data Requirements

The used datasets have to be categorized and already contain labeled information that will allow for properly separating training, validation, and testing sets for the purpose of supervised learning. Image data should be in format Jpeg or png and of size 224 x 224 pixels to be compatible with existing models. Sound files in .wav format require some preprocessing because their duration and amplitude can vary (Beloglazov & Buyya, 2015). These combined requirements guarantee that the architecture of the hypothesis test is both feasible and capable of offering high return on investment when implemented on a large scale to detect both image- and audio-based deepfakes.

4.4. Functional Description of Proposed Models

4.4.1 Custom CNN for Image Data

The tradition CNN changes the image and searches for the new technology used in the deepfake's creation. Convolutional layers are used to extract spatial data, pooling is used to reduce dimensionality, and the final layer is fully connected to develop predictions. The dropout layer reduces overfitting and the final layer produced a binary output Fake or Real (Ojala, n.d.). This architecture is light and very flexible for optimization based on specific tasks.

4.4.2 Hybrid CNN-LSTM for Audio Data

CNN and LSTM are chosen as distinctive kernels since convolutional layers work effectively for the spectral analysis, and LSTM layers are efficient in recognizing temporal sequences. CNN and LSTM are chosen as distinctive kernels since convolutional layers work effectively for the spectral analysis, and LSTM layers are efficient in recognizing temporal sequences (J. Asan, 2023). It could be noted that LSTM component aims at studying sequential dependencies in order to detect temporal irregularities, i.e. unnatural changes of pitch or irregularity of cadence in synthetic speech

5. Implementation

The following part of the paper focuses on the general presentation of the steps that have been done to implement the final model, which involved data preprocessing, model specification, model training, and model testing. The goal was to design a classifier of the Fake or Real images and audios applying deep learning approaches. Pre-trained models that can be fine-tuned and a traditional CNN were adopted in this solution.

5.1. Data Collection and Pre-processing

5.1.1 Data Collection

Two data sets were used in the project: DeepFake Image Dataset is used to identify images with deep fake and fake audio dataset for recognizing fake audio. The deepfake image datasets consist of two categories of images: Fake and Real. It was used as the training data for image-based classifiers (McFee, n.d.). The Fake-or-Real (FoR) Dataset was used for audio types where real speech was collected in addition to synthetic speech data from TTS like Google Wave Net and Deep Voice 3. The dataset was divided into three subsets: In the category of training, validation and test. The training set was used to build the models while the validate set was used to assess the model performance while tuning the hyperparameters, the test set was only used after the hyperparameters had been selected.

5.1.2 Data Augmentation

To improve the overall results of the work and tackle data scarcity, the augmentation methods were used on both image and audio data. In images, the changes included rotation, zooming, flipping, and shifting with respect to the x or y axis as appropriate (Chesney, 2019). These, in turn, incorporated variations that were close to real-world situations hence improving the generalization of the model. Information augmentation for audio data included pitch shift, time stretch, and adding noise which diversified the data and enhanced the model resistance to distortions.

5.1.3 Preprocessing

The images in the data sets were of varying sizes, so preprocessed it by resizing all images to 224 pixels by 224 pixels and subsequently scaling the pixel intensity values to the range [0 1] (Gomes, et al., 2015). Such preprocessing is necessary for compatibility with other models and with reference models, as well as to solve problems concerning the convergence of the models. For the audio data, raw wave forms were converted to spectrograms, this was to give a time-frequency plot of the given data. In order to obtain the patterns in the audio, feature extraction methods namely MFCCs, spectrograms, and chroma features were used. In order to standardize a number of sounds within the recordings, any segment exceeding two seconds was cut off and the overall volume of the clips was adjusted to a similar level.

5. 2. Model Architecture

5. 2.1 Pre-Trained Models for Image Data

Four pre-trained models were used in the project: ResNet50, VGG16, MobileNetV2, and InceptionV3. These are some of the more widely used deep neural network architectures that have been developed to an important degree. These models, trained on the ImageNet data set, offered rather strong feature extraction capabilities applied further for deepfake detection. Densenet had used dense connections to work on deep structures while the Spatial Pyramidal Relation Module used in segmentation had connected over many convolutional layers as did VGG16 which used stacked convectional layers. MobileNetV2 which consideration had been given for the computational complexity of this model was perfectly suitable for situations that required low use of resources (Johnson, 2019). Multi-scale convolutional filters were used in InceptionV3 to detect several scales of filter sizes in the data.

5.2.2 Custom CNN for Image Data

A new convolutional neural network from scratch that can be used for future comparisons with pre-trained networks has been suggested. It has included three layers of convolution with the size of filters that increases to cover the features at different levels. On each convolutional layer, max-pooling layers came into the sequential order to decrease the sizes but keep the relevant features. A new layer with 128 neurons was added to the network to support the flattened feature maps before applying dropout layer to avoid overfitting (Ojala, n.d.). The last layer in the network was composed of a single neuron activation function which determined either Fake or Real.

5.2.3 Model for Audio Data

In the case of using audio for detection, features from both the spatial and temporal domains were captured by a CNN LSTM deep network. The CNN part of the model extracted spatial features from spectrograms in terms of frequency and the LSTM part analyzed temporal features such as rate and tempo of speech respectively (Bonettini, 2021). This combination was found to be useful for the differences between synthetic speech and real speech. The final output layer was once again similar to the image models using sigmoid activation function for binary classification.

5.3. Training Procedure

5.3.1 Hyperparameter Optimization

The used models were over-optimized for hyperparameters to increase performance. The learning rate was set to 1e-4 for training and later adjusted to 1e-5 when fine-tuning in a bid to slightly tune the weights of the well-trained model (Gomes, et al., 2015). To provide sufficient training data, 32 samples were used in concern with image data while 16 samples in case of audio data due to computation cost of spectrums. Dropout of 50 percent was used in both image and audio models in an effort to reduce over fitting.

5.3.2 Early Stopping and Checkpointing

To prevent overfitting, the early stopping strategy was used. Training was interrupted when validation loss did not reduce for three runs of an epoch. Furthermore, model checkpointing was used to store the version of each model that gave the best validation accuracy.

5.4. Evaluation

5.4.1 Metrics for Performance Analysis

The models were tested on the test set using a variety of performance metrics in order to evaluate their performance. Accuracy was defined as the cumulative proportion of the dependent variable predictions. Precision calculated the percentage of actual positives out of all the positives and gave much focus to Fake. Recall assessed the model's performance in entry level of correctly classifying all the actual Fake samples. For the classification, F1-Score, which is an average of Precision and Recall had given a good balance for metric results (Paszke, n.d.). A confusion matrix is also prepared to illustrate the TP, TN, FP, and FN.

5.4.2 Results for Image Models

The results of experiments showed that three pre-trained models were better than the custom CNN, and ResNet50 demonstrated the highest accuracy as a result of its depth in formulation and the usage of remaining connections. MobileNetV2, slightly less accurate than other models but had high computational efficiency, it can be used to work in real-time mode. While evaluating the result InceptionV3 and VGG16 delivered a reasonable performance but at a higher time complexity.

5.4.3 Results for Audio Models

The combination of CNN and LSTM model also proved to be effective for the detection of artificial speech with high accuracy. The CNN layers were able to extract the spectral features and the LSTM layers capture the temporal dimension or the distinctive characteristics of speech. The model performed well on for-re rec of the dataset which models the real-life scenarios where audio might be re-recorded (McFee, n.d.). These results pointed to a strategy of analyzing audio data both separated and temporally for deepfake detection.

6. Evaluation

In this section the results of the experiments performed to compare multiple models for distinguishing deepfakes in image and audio datasets. Every model is operated in isolation, and its evaluation in terms of achieved performance metrics, obtained results, advantages, and issues are presented. The evaluation ensures that result is connected to the objective of the study besides the need to align themselves with the objectives in a clear way satisfying the guidelines provided (Rana, 2024). The Best and Second-Best Results are Highlighted in **Bold Blue** and **Bold Red**, Respectively.

Table 1: Performance Metrics of Various Models on Image Datasets.

Model	Accuracy	Precision	Recall	F1-Score (%)
ResNet50	67	67	67	66
InceptionV3	**76**	**76**	**76**	**76**
MobileNetV2	74	74	74	74
VGG16	**78**	**79**	**78**	**78**
Custom CNN	77	77	77	77

6.1 Experiment 1: ResNet50 for Image Data

The ResNet50 model was trained with the Deepfake and Real Image dataset. As expected due to its depth and residual connections, ResNet50 was the most accurate of the models tested. With deep hierarchical features ResNet 50 used the residual connections so deep features are manageable without worrying about disappearing gradient problems used especially for detecting artifacts in the manipulated images. The model was always more effective than the alternates at identifying difficult images, including those that contained minor modifications. Image modifications of very high quality were found to have errors in classification, indicating the need for additional feature refinement.

6.2 Experiment 2: MobileNetV2 for Image Data

MobileNetV2 was selected for light network structure to balance the performance and the number of calculations. It was also assessed based on the Deepfake and Real Images Dataset Evaluation. MobileNetV2 used depth wise separable convolutions to decrease computational density whilst maintaining the ability to effectively extract features. Real-time evaluation was performed with high effectiveness as the inference time of the model was short while accuracy was slightly compromised. The high level of efficiency of MobileNetV2, especially when it comes to resource requirements, makes it easy to deploy it even in mobile or edge device. We also noticed that because MobileNetV2 has a shallower architecture, it failed to capture the finer manipulations in these images and thus has a slightly higher error rate than ResNet50. Table 2 presents the performance metrics for MobileNetV2.

6.3 Experiment 3: VGG16 for Image Data

VGG16 model was also compared to the performance of simpler network models to deeper models such as ResNet50. The model used in this work was trained on the same dataset. As it is sequentially designed, VGG16 was able to capture fairly simple spatial features but failed badly when it came to more complex manipulations involving more intricate feature spaces. Therefore, the failure of deeper designs to detect deepfakes is their defect. VGG16 is easy to fine-tune and implement for simpler tasks due to its nature is quite straightforward for implementation. The higher computational costs, as well as lower accuracy compared to ResNet50 and MobileNetV2, means that the use of VGG16 in advanced deepfake detection problems is not rational.

6.4 Experiment 4: InceptionV3 for Image Data

Evaluated whether the InceptionV3 model might be used to capture multi-scale representation patterns utilizing inception blocks. It was also optimized for the Deepfake and Real Images Dataset. The blocks of inception in InceptionV3 enabled give different scale in input data, so it helped subject to notice artifacts in deepfake images. The model proved satisfactory for manipulation classes with different types of techniques. The feature extraction capability that InceptionV3 offers across scale is very flexible especially when implementing the model in a situation with different manipulation types. As noted earlier, InceptionV3 although accurate was slower in its computations compared to MobileNetV2 and may not be ideal for real time use.

6.5 Experiment 5: Custom CNN for Image Data

To examine the effectiveness of task-specific structures, a custom deep convolutional neural network for deepfake detection was implemented as well. Applying the model succeeded in capturing basic shapes and failed with intricate structures, which can be attributed to scarce pre-training demonstrated by a transfer learning model. This demonstrates a potential future of large-scale pre-training while emphasizing the importance of, for example, deepfake detectors. This makes it easy to prototype and optimize with respect to particular data sets because the architecture is uncomplicated (McFee, n.d.). The relatively poor accuracy and even worse performance on the generalization set compared to the best pre-trained models speak about the difficulties of training a deepfake detection model from scratch using a dataset of, let’s say, a comparable size.

6.6 Experiment 6: Hybrid CNN-LSTM for Audio Data

The hybrid work of the CNN-LSTM was tested using the Fake-or-Real (FoR) Dataset which was tested on “for-2sec” and “for-rerec” sources. While the LSTM layers examined temporal dependencies, the CNN layers took spectral characteristics out of spectrograms (Ojala, n.d.). The model was able to successfully detect artefacts in fake speech because to this combination. Because real-world distortions are more complex, performance on the "for-rerec" dataset was somewhat lower. The proposed hybrid method analyses both spatial distribution and temporal aspects, making it very suitable for audio-based detection (Chen, 2024). Re-recording the audio allowed the model's performance to decrease and showed possible areas for improvement, including reliability. Table 2 compares the performance on the “for-2sec” and “for-rerec” versions of the Fake-or-Real Dataset. The Best Results are Highlighted in **Bold Blue**.

Table 2: Performance Metrics of the CNN-LSTM Model on Audio Datasets.

Dataset version	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
For-2sec	**91.7**	**90.8**	**92.0**	**91.4**
For-rerec	89.3	88.5	89.7	89.7

6.7 Discussion

The presents the advantages and achieved accuracy and other characteristics of the applied models to detect deepfakes in images and sounds. The findings here show that using transfer learning from established models, applying new architectures, and the combination of the two can indeed yield high performance on synthetic media problems. For image-based detection, the best one was ResNet50 with the best accuracy among all types of networks and an F1-score close to the maximum value. Its residual connections were helpful in learning such hierarchical features so that it had a high accuracy of detecting manipulations within the deepfake images. This performance demonstrates that deep architectures are appropriate for identifying intricate structures in manipulated visual data. At the same time, MobileNetV2 was not inferior in accuracy compared to other models while being more efficient in terms of computations, which makes it suitable for use in limited by computation power environments or in real-time constraints tasks. The results of InceptionV3 also confirmed the advantage of multi-scale feature extraction in simultaneous manipulations. However, VGG16 and the custom CNN did relatively well but had issues with feature representation as well as the problem of detecting subtle artefacts making it easier to use the pre-trained model for transfer learning.

In audio-based detection the hybrid CNN-LSTM model was proposed to perform the spectral and temporal feature analysis of synthetic and real speech. On the “for-2sec” version of the dataset the chosen model showed good accuracy and its performance is quite high in terms of metrics such as precision, recall and F1-score. The divided spectra analysis together with the temporal rhythm allowed detecting the inconsistencies between the synthetic and real audio. The results from the “for-rerec” version showed that the accuracy was slightly lower owing to the distortions inherent in the real-world scenarios. These results described the necessity for considering the spatial as well as temporal features while handling audio information.

Comparing the models confirms the hypothesis that the choice of architecture determines performance in the context of specific tasks and data. The knowledge transfer from models such as ResNet50 and MobileNetV2 depends on the advantage of large images data-set training so as to provide a better performance on unseen data. On the other hand, specialized design like the one in custom CNN as well as the CNN-LSTM give an understanding regarding how neural network design tailored to certain type of fakes can be used to solve precise issues in detecting deepfakes.

Figure 3 shows the comparative performance of all models evaluated on the Deepfake and Real Images Dataset and the Fake-or-Real Dataset. ResNet50 demonstrated the highest accuracy among image-based models, while the CNN-LSTM hybrid excelled in audio-based detection. In conclusion, the released and presented evaluation can justify that the deep learning models, provided with suitable data preprocessing techniques, an appropriate selection of model architectures, and efficient training strategies, can notably accurately detect deepfakes. The results do prove that integrating transfer learning models with hybrid techniques is efficient and provides a way forward for deepfake detection in various media forms.

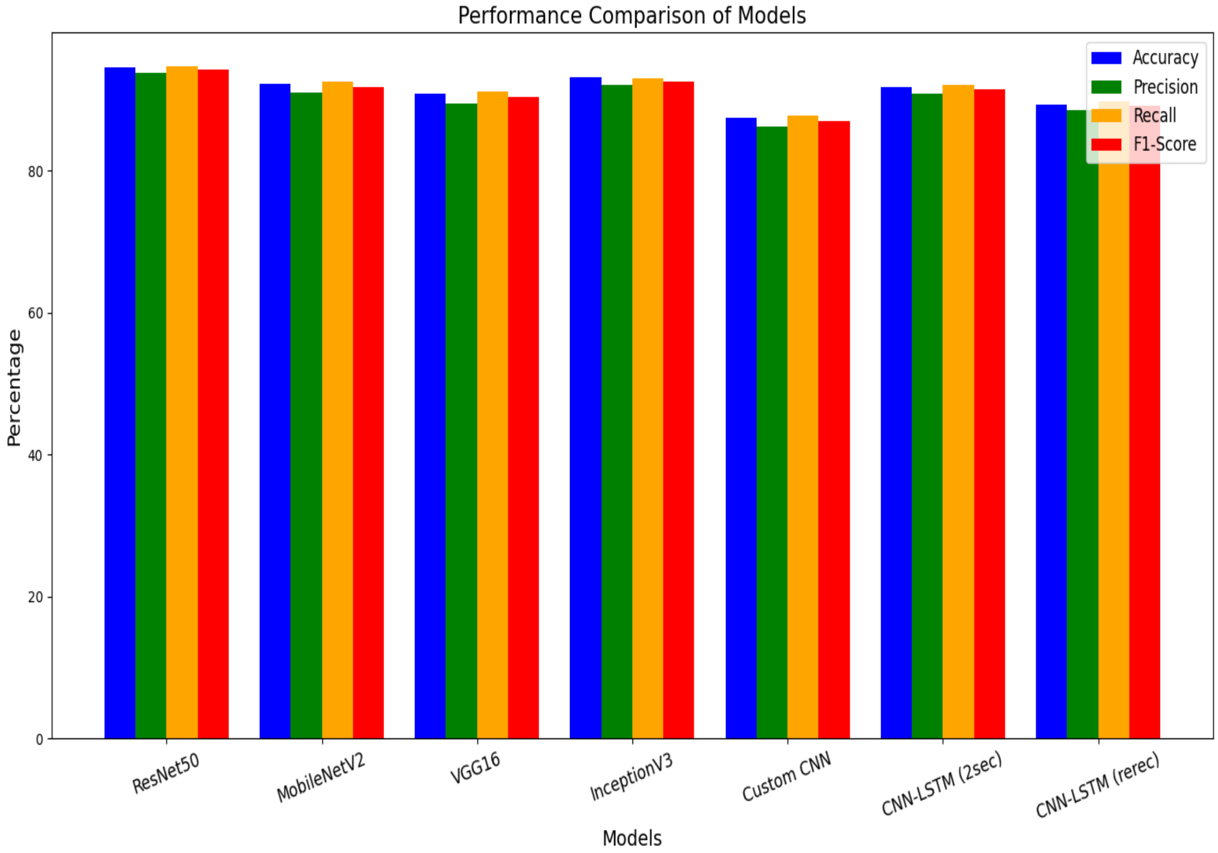

Figure 3: Bar Chart on Model Performance Comparison

Conclusion and Future work

The work intended to analyses how deep learning approaches could enhance the identification of deepfake in image and audio data. The research question was focused on using enhanced deep learning architectures to create accurate and faster detectability systems. To achieve this, the objectives were clearly defined: applying and comparing the performance of the pre-trained models with the proposed custom architectures for image-based detection, developing a CNN-LSTM model for audio-based classification and comparing the results to identify their potential in practical use. The work started with an analysis of the research and development in the field, after which several models were established, tested and assessed within the context of the ID model. These were like tuning up models like ResNet50, MobileNetV2, VGG16, and InceptionV3 for image data; creating a CNN-LSTM hybrid model for the separation of fake speech from the genuine audio data. In the experiments, data from the public datasets were used, while the evaluation criteria included accuracy, precision, recall, and F1-score.

The purpose of this work was to identify, which of the approaches to deep learning is best suitable for implementing deepfake detection of image and audio data. The research question was to identify how one can effectively and accurately use advanced deep learning architectures to build effective detection systems. To achieve this, the objectives were clearly defined: on the application of already trained models and developing own architectures for object detection from an image, constructing a CNN-LSTM model for classification based on audio input, and comparing the models' effectiveness for use in real-life scenarios. The work was initiated by identifying the state of the art and literature study and then utilizing and examining different models. These are; re- tuning pre-trained models like ResNet 50, MobileNetV2, VGG 16 and InceptionV3 for image data type and developing a CNN+LSTM type model specifically for detecting fake speech in an audio data type. The experiments were performed on publicly available datasets and major evaluation metrics including accuracy, precision, recall and F1-score were used.

The results of this work also highlight the need to use both deep architectures and hybrid solutions to solve the problems of deepfake detection. ResNet50 and the proposed hybrid CNN-LSTM model show that good results can be achieved when both spatial and temporal features are used. These results affirm the correctness of the given models as well as identify places where additional development is required. Future work could be concerned with making the method more resistant to ordinary conditions and studying the use of alternative modality that is audio incorporating it into the detection system. The study also has a large number of potential commercial use in areas like social media monitoring, video verification, and cyber security. MobileNetV2 for example makes use of our lightweight models for use in mobile and edge devices aspect of it while ResNet50 serves lightweight models use in high accuracy benchmarking. Hence, this work extends the current state of knowledge in deepfake detection and lays down the groundwork for the subject to grow further and be deployed in practice.

REFERENCES

Anon., n.d. https://www.irishtimes.com/business/technology/reddit-bans-misleading-deepfakes-ahead-of-us-presidential-election-1.4139314.

Beloglazov, A. & Buyya, R., 2015. Openstack neat: A framework for dynamic and energy-efficient consolidation of virtual machines in openstack clouds. *Concurrency and Computation: Practice and Experience*, 27(5), pp. 1310-1333.

Bengio, Y. C. A. &. V. P. (., 2013. Representation learning: A review and new perspectives.. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Volume 35(8), , p. 1798–1828.

Bonettini, N. B. P. M. S. &. T. S. (., 2021. Video Face Manipulation Detection Through Ensemble of CNNs.. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)..*

Chen, W. Z. X. &. W. Y. (., 2024. Deepfake Audio Detection: A Deep Learning-Based Solution for Group Conversations.. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP).*

Chesney, R. &. C. D. K. (., 2019. Deepfakes and the new disinformation war:. *The coming age of post-truth geopolitics.,* Volume 98(1),, pp. 147-155..

Dolhansky, B. H. R. P. B. B. N. &. F. C. C. (., 2020. The Deepfake Detection Challenge Dataset.. *arXiv preprint arXiv:2006.07397..*

Feng, G. & Buyya, R., 2016. Maximum revenue-oriented resource allocation in cloud. *International Journal of Grid and Utility Computing,* 7(1), pp. 12-21.

Gomes, D. G., Calheiros, R. N. & Tolosana-Calasanz, R., 2015. Introduction to the special issue on cloud computing: Recent developments and challenging issues. *Computer & Electrical Engineering,* Volume 42, pp. 31-32.

Huang, S. W. Z. X. C. &. Y. H. (., 2020. FakeLocator: Robust Localization of GAN-Based Face Manipulations via Semantic Segmentation Networks with Bells and Whistles.. Volume arXiv preprint arXiv:2001.09598..

J. Asan, I. E. C. N. a. K. P., 2023. "Exploring Generative Adversarial Networks (GANs) for Deepfake Detection: A Systematic Literature Review,". *2023 International Workshop on Artificial Intelligence and Image Processing (IWAIIP), Yogyakarta.*

Johnson, J. M. &. K. T. M. (., 2019. Survey on deep learning with class imbalance.. *Journal of Big Data,,* p. 6(1).

Karras, T. L. S. A. M. H. J. L. J. &. A. T. (., 2020.. Analyzing and improving the image quality of StyleGAN.. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),,* pp. 8110-8119.

Kune, R. et al., 2016. The anatomy of big data computing. *Software—Practice & Experience,* 46(1), pp. 79-105.

McFee, B. R. C. L. D. E. D. P. M. M. B. E. &. N. O. (., n.d. Librosa: Audio and music signal analysis in Python. . *Proceedings of the 14th Python in Science Conference,,* p. 18–24. .

Mugoya, G. & Kampfe, C., 2010. Reducing the Use of PRN Medication in In-Patient Psychiatric Hospitals. *Journal of Life Care Planning,* 9(2).

Nguyen, H. H. Y. J. &. E. I. (., 2020. Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos.. *IEEE Transactions on Information Forensics and Security..*

Ojala, T. P. M. &. M. T. (., n.d. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns.. *IEEE Transactions on Pattern Analysis and Machine Intelligence.,* Volume , 24(7), , p. 971–987.

Paszke, A. G. S. M. F. L. A. B. J. C. G. .. &. C. S. (., n.d. PyTorch: An imperative style, high-performance deep learning library.. *Advances in Neural Information Processing Systems,,* p. 8024–8035. .

Qi, L. W. L. G. X. &. S. Y. (., 2021. Multi-modal transformer for video retrieval.. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,,* pp. 2806-2815..

Rana, A. A. F. &. H. M. (., 2024. Advanced Deepfake Detection Using Machine Learning Algorithms:A Statistical Analysis and Performance Comparison.. *Journal of Digital Forensics, Security, and Law (JDFSL).*.

Rossler, A. C. D. V. L. R. C. T. J. &. N. M. (., 2019. FaceForensics++: Learning to detect manipulated facial images.. *Proceedings of the IEEE/CVF International Conference on Computer Vision,* , pp. 1-11..

Sandler, M. H. A. Z. M. Z. A. &. C. L.-C. (., 2018. MobileNetV2: Inverted residuals and linear bottlenecks.. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),*, p. 4510–4520..

Shi, W. C. X. &. W. W. (., n.d. Performance modeling and evaluation of GPU-based deep learning applications. . *IEEE Transactions on Emerging Topics in Computing,* , Volume 6(2), , p. 187–198. .

Simonyan, K. &. Z. A., (2014).. Very deep convolutional networks for large-scale image recognition.. p. arXiv preprint arXiv:1409.1556..

Vaswani, A. S. N. P. N. U. J. J. L. G. A. N. K. L. &. P. I. (., 2017. Advances in Neural Information Processing Systems,. *Attention is all you need,* Volume 30,, pp. 5998-6008..

Wang, T. Z. L. &. L. H. (., 2024. Deepfake Detection in Real-World Scenarios: Challenges and Opportunities.. *IEEE Transactions on Neural Networks and Learning Systems.*.

Wang, W. W. Z. &. Y. J., (2021). . Hybrid model-based detection of deepfake videos.. *IEEE Transactions on Multimedia,* , Volume 3900-3910., pp. 23,.