# DEEPFAKE DETECTION MODEL USING DEEP LEARNING TECHNIQUES

Thesis
Submitted in partial fulfillment of the requirements for the degree of

MASTER OF TECHNOLOGY
in
INFORMATION TECHNOLOGY

by
SOURABH MAKHIJA
(212IT028)

DEPARTMENT OF INFORMATION TECHNOLOGY
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE - 575025

JUNE, 2023

# DECLARATION

I hereby *declare* that the Report of the P.G. Project Work entitled "DEEPFAKE DETECTION MODEL USING DEEP LEARNING TECHNIQUES", which is being submitted to National Institute of Technology Karnataka Surathkal, in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Information Technology in the Department of Information Technology, is a *bonafide report of the work carried out by me.* The material contained in this Report has not been submitted at any University or Institution for the award of any degree.

Sourabh Makhija (212IT028)
(Signature of the Student with Date)
Dept. of Information Technology
NITK Surathkal, Mangalore

Place: NITK, Surathkal
Date: 20-06-2023

# CERTIFICATE

This is to *certify* that the P.G. Project Work Report entitled “DEEPFAKE DETECTION MODEL USING DEEP LEARNING TECHNIQUES”, submitted by Sourabh Makhija (212IT028) the record of the work carried out by him, is *accepted as the P.G. Project Work report submission* in partial fulfillment of the requirement for the award of degree of Master of Technology in Information Technology in the Department of Information Technology.

Dr. Dinesh Naik
(Guide Signature with Date and Seal)

Chairman - DPGC
(Signature with Date and Seal)

# ACKNOWLEDGEMENT

I have taken efforts in this project, however, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Dr.Dinesh Naik for his guidance and constant supervision as well as for providing necessary information regarding the project also for his support in completing the project. I would like to express my gratitude towards all members of National Institute of Technology Karnataka Surathkal, Information Technology Department for their kind cooperation and encouragement which helped me in completion of this project.

I also reveal my thanks to all my classmates and my family for constant support.

Sourabh Makhija

# ABSTRACT

The widespread usage of face swapping deepfake models, which creates a considerable number of phoney videos and images that seem incredibly real, is presently putting the privacy of people and whole nations in danger. Fake news is sweeping the internet like wildfire because DeepFake production tools have developed so quickly and because so many researchers and companies are interested in exploring their limits. This research suggests EfficientNet CNNs with an LSTM classifier as a DeepFake detection and classification model. After the model has recovered the face part from video frames, the CNN enhanced with LSTM & ReLU is utilised to extract features from these faces and classify. For the Deepfake video detection, a CNN equipped with LSTM model was applied to ensure model validity while keeping an appropriate weight.

Therefore, in order to effectively detect deep fakes, requires stronger deep fake detection algorithms. The system that has been proposed is built on a combination of CNN and RNN. The system that is being presented makes use of the CNN model to extract feature vectors from the movies, and then it uses those feature vectors to train an RNN model to classify the videos as Real or deep-fake.

The proposed model's effectiveness was evaluated on the Face Forensics and Celeb-DF deep fake datasets. Experimental results demonstrated that the suggested model had an average prediction rate of 84.61% for deep-fake films and 90.22% for Face Forensics movies under real network diffusion settings. In comparison to convolutional neural network-based systems like ShuffleNet, DenseNet, and MobileNet, the suggested model produced the best results with an accuracy rate of 88.46%.

***Keywords***— Computer Vision, Deep Neural Network, EfficientNet, ResNet, XceptionNet

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ABBREVIATIONS

| | |
|---|---|
| AI | Artificial intelligence |
| CNN | Convolutional Neural Networks |
| Celeb-DF | Celeb Deep-Fake |
| DFDC | Deep Fake Detection Challenge |
| DNN | Deep Neural Network |
| GANs | Generative Adversarial Networks |
| GPU | Graphic Processing Unit |
| GRU | Gated Recurrent Unit |
| FP | False Positive |
| FPS | Frames per second |
| FN | False Negative |
| ILSVRC | ImageNet Large-Scale Visual Recognition Challenge |
| LRCN | Long-term Recurrent Convolution Network |
| LSTM | Long Short-Term Memory |
| MT-CNN | Multi-task Cascaded Convolutional Networks |
| NLP | Natural Language Processing |
| PPG | Photoplethysmography |
| ResNet | Residual Network |
| RNN | Recurrent Neural Network |
| SVM | Support Vector Machine |
| TP | True Positive |
| TN | True Negative |
| ViT | Vision Transformer |

# Chapter 1

# INTRODUCTION

AI algorithms have recently made it easier and more enhanced & realistic to produce what are known as Deep Fake videos and images. This has raised concerns about intentionally spreading false news. These techniques have the potential to create manipulated recordings of events or statements by well-known celebrities, which can mislead large audiences in dangerous ways. One of the contentious topics in discussions about security measures in different systems is the DeepFake network. The main challenge has been finding ways to quickly and accurately determine face similarity or matches, despite recent advancements in facial reconstruction. Traditional analysis methods used to detect image forgeries are often inadequate for video forensics due to lossy compression and significant data degradation. Therefore, real-time Deepfake facial reconstruction for security purposes is challenging due to limited technology and efficiency. To evaluate the performance of the proposed model, several Deepfake datasets were used.

DeepFakes are rendered audibly or visually, usually as videos, that have been made artificially. These frequently made without the subject's consent videos can be used to defame influential people or influence public opinion. In a court of law, an audio or video recording could be used as uncontested evidence. An attacker can produce such precise renderings with GANs, which were examined by the authors in (Goodfellow et al. 2020), by using a regular desktop computer with an aftermarket GPU. They are easily trickable, both by robots and by humans. Recent advances in Deepfake face-based altering methods have made it possible to swap out one person's face for another Tolosana et al. (2020).

Therefore, it seems extraordinary to use supervised learning in addition to a copy-move modification to automatically swap out one person's face for another. Now it's possible to animate a clear set and turn it into a series of video frames. So even a monument can suddenly come to life thanks to technology Đorđević, Milivojević & Gavrovska (2020).

Using a model known as the GAN, DeepFake substitutes a different person's face for the facial traits that are actually visible in real-world footage. It is possible to construct realistic faces that can be retrieved and clipped into the original movie in a way that seems practically perfect because GAN models were developed using hundreds of photos (Goodfellow et al. 2020). Through appropriate post-processing or post-production processing, this resulting video can offer a better level of authenticity. Before the introduction of fake videos, the writers Zhang & Zhao (2020) thought that videotapes were typically reliable and trustworthy, and this was in interactive media forensics, which is frequently employed as hard evidence.

On the other side, the rise of DeepFake videos is shattering people's confidence. There is growing fear that this technology could be abused and have a huge impact on people's life once it is used as evidence in court, the media and publishing, diplomatic elections, and television and infotainment. Even some people think that this kind of technical growth might hinder society's progress. Therefore, identifying and detecting such phoney videos is crucial, whether it is for official or illegitimate purposes Lu et al. (2021).

As manipulations become more convincing, public figures may be placed in fictional scenarios, creating a sense that they can be made to say anything. Video evidence may lose its credibility as a means of confirmation, potentially leading the public to lose faith in all media. This underscores the importance of reliable multimedia sources in mainstream media to cater to the general audience. To identify Deepfakes in video, several algorithms have been created.*DeFaking Deepfakes: Understanding Journalists' Needs for Deepfake Detection* (2020) study demonstrated that although some of these techniques have shown some degree of effectiveness, the majority of these algorithms have also failed when tested against external data acquired from environments other than their study conditions.

The current widespread utilization of face-swapping Deepfake algorithms presents a significant danger to the privacy of individuals and nations due to the production of a high volume of genuine-looking fake videos and images. Due to the negative consequences Deepfake films have on society, being able to distinguish between them and real films has become a critical concern. The significant progress made in the development of GANs and other production technologies has resulted in the production of convincing fake media that could potentially have a harmful impact on society Heo et al. (2021). However, the effectiveness of existing Deepfake detection systems is being surpassed by the advancements in generation techniques, which highlights the need for an improved Deepfake detector that can analyze media created through any method. The development of a Deepfake video or image detector can be generally applied to various innovative techniques that have been recently introduced in challenging datasets. Awotunde et al. (2023)The main objective of this research was to create a system that can surpass the results achieved by current cutting edge techniques using several measures of performance.

DeepFake detection systems often use multi-modal detection algorithms to identify whether the target material has been altered or synthesised. The development of AI-based algorithms for algorithmic detection strategies, such as the Vision Transformer Heo et al. (2021), MesoNet Afchar et al. (2018), which the authors recommended, and the two-stream neural network Zhou et al. (2017), among others, is a common focus of current detection methods. On the other hand, manual image processing is given less thought in favour of emphasising the important portions of an image Tran et al. (2021). The processing of all the movies frequently results in the model becoming bulkier.The DeepFake detection method was enhanced by combining a DL-based model and a human processing technique in this research. The CNN-based model receives carefully selected and processed crucial data, regions, and characteristics. These networks train more effectively and more accurately by concentrating on the most important pieces of information.

Using neural network techniques like GANs or auto encoders, among others, deep fake is a technique for creating human photographs. These technologies may overlay target images onto source videos using deep learning techniques to create deep fake movies that look real. There is no way to tell these videos apart with the naked eye because they are so convincingly manufactured. This study presents a new method that utilizes deep learning to distinguish between fake videos generated by AI and genuine videos. The approach takes advantage of the limitations of deep fake technology to differentiate between authentic and artificially produced footage.

Although the human eye may not be capable of detecting certain recognizable artifacts in frames produced by existing deep fake techniques, trained neural networks are able to do so. The use of Res-Next Convolution Neural Networks allows for successful identification of the distinct artifacts caused by deepfake creation techniques. This method involves extracting features at the frame level and using them to train an LSTM-based RNN to determine whether a video has been manipulated, such as in the case of a deep fake.

There are numerous instances where deepfake creation technology is used to deceive users of social media by disseminating fake deepfake videos of celebrities like Mark Zuckerberg from the House A.I. Hearing, Donald Trump from the Breaking Bad series, where he played James McGill, Barack Obama from the PSA, and many others. The necessity to precisely identify these deepfakes so that they can be separated from authentic films derives from the fact that these deepfakes cause extreme panic among average people Tran et al. (2021). The discipline of video manipulation has altered as a result of recent technological advancements. The paradigm shift has been fueled by improvements made to contemporary open source deep learning algorithms like TensorFlow, Keras, and PyTorch as well as affordable access to powerful computing resources.

Additionally, the availability of these pretrained models via smartphones and desktop programmes like FaceApp and Face Swap has rendered the deepfake production a silly thing. These programmes produce a synthetic face makeover of real videos that is incredibly lifelike. Additional features offered by these apps include the ability to alter the user's face, hair, gender, age, and other characteristics Ciftci & Demir (2019). The user of these apps can also produce deepfakes that are of extremely high quality and are virtually undetectable. Even while there are some malicious deepfake videos, they are still in the minority. The technologies that have been made public and that produce deepfake films are primarily used to produce fake celebrity pornography or revenge porn Li, Chang, Farid & Lyu (2018). Nude videos of Brad Pitt and Angelina Jolie are a couple of instances. Celebrities and other famous people are the subject of pornographic content, fake surveillance films, false news, and malicious hoaxes because deepfake videos have a realistic appearance. Political animosity is frequently stoked by the Deepfakes Güera & Delp (2018). Because of this, it is crucial to identify deepfake films and prevent their spread across social media sites.

- Studied the behaviour of EfficientNet family with varying number of parameters and accuracy.
- Implimented highly accurate EfficientNet model with LSTM for further classification and compared its variation with basic model.
- Similarly implimented backbone CNN model with LSTM and did an experimental analysis for MobileNet, DenseNet & ShuffleNet.

## 1.1 MOTIVATION

To overcome these difficulties, the suggested model was tested against a sizable dataset of films that featured realistic alterations. This was done to guarantee that the system operates effectively and efficiently. The model is also used to quickly identify whether a video is a Deepfake or not. The main goal of the proposed model is to show how to simplify the Deepfake detection issue by using a few of the most popular classification methods to identify fake films. Prudent model selection will increase the ability to address the Deepfake detection issue because the present classification models are constructed for high accuracy.

Particularly frequently, it is believed that the look and actions of people (mainly humans) in pictures and films serve as reliable proof of actual events. Modern technologies make manipulating these media assets considerably simpler and more accessible, despite the fact that this impression is gradually changing. This gap poses a risk to society when distorted media are spread over social networks and ingested by a public that lacks the knowledge to confirm their reality. The need for video forgery detection is very critical.

# Chapter 2

# LITERATURE REVIEW

## 2.1 BACKGROUND AND RELATED WORKS

In order to detect artefacts, Face Warping Artefacts Li & Lyu (2018) utilized a CNN model to compare the generated facial regions and their surroundings. In this experiment, two categories of facial distortions were observed. The researchers devised their approach after recognizing the limitations of the existing deepfake algorithm, which can only generate images with a specific resolution. Further adjustments are required to align these images with the faces intended to replace in the source video. However, their methodology did not consider the temporal analysis of the frames.

A new technique for identifying deep fakes is described in Detection by Eye Blinking Li, Chang & Lyu (2018) by the eye blinking being a key factor in determining whether a movie is deep fake or authentic. The cropped frames of eye blinking were temporally analysed using the LRCN. As deepfake creation algorithms have become increasingly sophisticated, the absence of eye blinking alone cannot serve as a reliable indicator of a deepfake. In order to detect such fakes, other factors such as teeth appearance, facial wrinkles, and incorrect brow positioning must also be considered.

Capsule networks are utilized in various scenarios to identify counterfeit images and videos, including detecting replay attacks and computer-generated videos. The methodology is discussed in the article "Capsule Networks for the Detection of Fake Images and Videos." Nguyen, Yamagishi & Echizen (2019). The use of Capsule Networks for deepfake detection offers several advantages. The vector-based representation of visual elements allows Capsule Networks to capture more nuanced relationships between features, making them potentially more sensitive to deepfake artifacts. Furthermore, the reconstruction component of Capsule Networks helps regularize the model and enhance its ability to distinguish between real and manipulated videos. It should be pointed out that the detection of deepfakes using Capsule Networks is an area of research that is constantly evolving, and the effectiveness of these models is heavily dependent on the availability of diverse and high-quality training datasets. Scientists are constantly experimenting with various architectures and training methods to enhance the accuracy and resilience of Capsule Networks in detecting deepfakes.

The approach of applying RNN for sequential processing of the frames combined with the pre-trained ImageNet model was employed for RNN for deepfake detection Güera & Delp (2018). They used the HOHO Laptev et al. (2008) dataset, which consists of only 600 videos. The dataset provided contains only a small number of films of a similar nature, which might pose challenges for achieving optimal performance on real-time data. To address this, we plan to incorporate a significant amount of real-time data during the model training process.

On pristine and deepfake portrait video pairs, the synthetic portrait videos utilising biological signals Awotunde et al. (2023) technique extracts biological signals from facial regions. Applied transformations to capture the signal properties in feature vector and PPG maps, compute the spatial coherence and temporal consistency, and then train a probabilistic SVM and a CNN. The classification of the video as a deepfake or a pristine is then made using the average of authenticity probability.

False Catcher is effective at detecting false content, regardless of who created it, the content itself, or the quality of the video. However, creating a differentiable loss function that adheres to the recommended signal processing steps is challenging because the lack of a discriminator has resulted in the loss of biological signals.

According to Nguyen, Fang, Yamagishi & Echizen (2019), DeepFake detection is considered as a binary classification problem, wherein the detection approaches' capability to differentiate between original and DeepFake videos was assessed. The technique mainly relied on the quality of image measures, with an SVM classifier that can detect high-quality DeepFake videos with an equal error rate of 8.97.%. The system needs a more reliable detection algorithm, and one of its limitations is that it relies on subjective assessments to determine how vulnerable human subjects are to DeepFakes. The LAIR dataset was created by Girgis et al. (2018), these word arrays were improved by utilizing them to enhance the data supplied to the language model, which were later fed into the deep learning models. To achieve the best results, this approach requires to merge the GRU and CNNs to boost detection accuracy. Montserrat et al. (2020) developed a strategy that is both straightforward and effective. The current approach utilizes a combination of CNNs, RNNs, and a specific dataset to achieve the best possible results. The system primarily concentrates on identifying facial modifications and does not consider the analysis of audio content. In future research, incorporating audio analysis could significantly enhance the accuracy of detection. Moreover, the system efficiently processes videos by utilizing a single GPU to ensure fast processing speed.

In *DeFaking Deepfakes: Understanding Journalists' Needs for Deepfake Detection* (2020) Incorporating intriguing ideas or early iterations of the usual option tools with qualitative interviews. It laid the groundwork for the development of DeepFake detection techniques. There is no definitive approach or process to use when creating the aforementioned tool. Wodajo and colleagues (2021) Lu et al. (2021) ViT and CNN. The authors successfully achieved an acceptable outcome by implementing a CNN layer into the ViT framework for the DFDC dataset. By combining new information from the Deepfake study, this approach could be improved upon to become broader, more accurate, and more resilient. Güera and colleagues (2018) Güera & Delp (2018) RNN and CNN The system uses a straightforward pipeline architecture and nonetheless produces a competitive outcome. The proposed system can investigate during training how to increase the effectiveness of the system against fake movies utilising nefarious approaches.

## 2.2 OUTCOME OF LITERATURE REVIEW

- In the recent times, there is a huge emergence of deep learning-based tools that are used to create believable manipulated media known as Deepfakes. These are realistic fake media, that can cause threat to reputation, privacy and can even prove to be a serious threat to public security.

- These can even be used to create political distress, spread fake terrorism or for blackmailing anyone. As with growing technology, the tampered media getting generated are way more realistic that it can even bluff the human eyes. Hence, the need of better deepfake detection algorithms for efficiently detect deepfakes.

- The Literature Survey helped to identify various approaches used for the different generations of dataset using machine learning models and different feature extraction methods.
- State of the art deep learning models like Xception-Net, Efficient net and modified ResNet models were not explored much.
- Third generation dataset like DFDC were being less efficient with machine learning model and earlier used deep learning architecture like VGG net.

## 2.3 PROBLEM STATEMENT

Develop a deep-fake detection model with the help of CNN over third generation dataset.

## 2.4 RESEARCH OBJECTIVES

- Design and implement a deep-fake detection model using Xception-Net CNN with classifier.
- Find out the best performing EfficientNet from its family over the decided dataset.
- Implement various backbone CNN with a classifier similar to base paper for classification over 2 kinds of dataset.
- To extensively evaluate the proposed approaches and benchmark against existing works.

# Chapter 3

# PROPOSED METHODOLOGY

This section discusses the suggested method for distinguishing fake videos and photographs from authentic ones. Figure 3.1 represents the flowchart of the proposed methodology. The proposed method involves cropping faces from video frames, which are then utilised to recognise modified recordings.After the face has been preprocessed and retrieved from the video, the image is fed into a deep learning model. It strengthens the deep learning model of the entire process. Based on the output of the deep learning model, this is how it determines whether anything is fake or real. Cropped faces are used to construct feature maps in the foundational CNN model. And then it is sent to the LSTM layer for further classification.

## 3.1 ARCHITECTURE

Deepfake detection using a combination of ResNet and LSTM classifier involves leveraging the ResNet architecture for feature extraction and utilizing a LSTM network for sequence analysis and classification. Here's an explanation of how this approach can be employed for deepfake detection:

ResNet Feature Extraction: ResNet is a deep convolutional neural network architecture that allows for training very deep networks by utilizing residual connections. These connections enable the network to effectively learn and extract hierarchical features from input images or frames. The pre-trained ResNet model can be utilized as a feature extractor for both real and deepfake videos.

Video Frame Extraction: Deepfake videos are typically composed of a sequence of frames. The first step is to extract individual frames from the video. These frames serve as the input for subsequent analysis using the ResNet-LSTM model.

ResNet Feature Extraction: Each frame extracted from the videos is passed through the pre-trained ResNet model. The ResNet model extracts high-level features from each frame, capturing discriminative information specific to deepfake manipulation. The output of the ResNet model can be a feature vector or a tensor representing the frame's visual content.

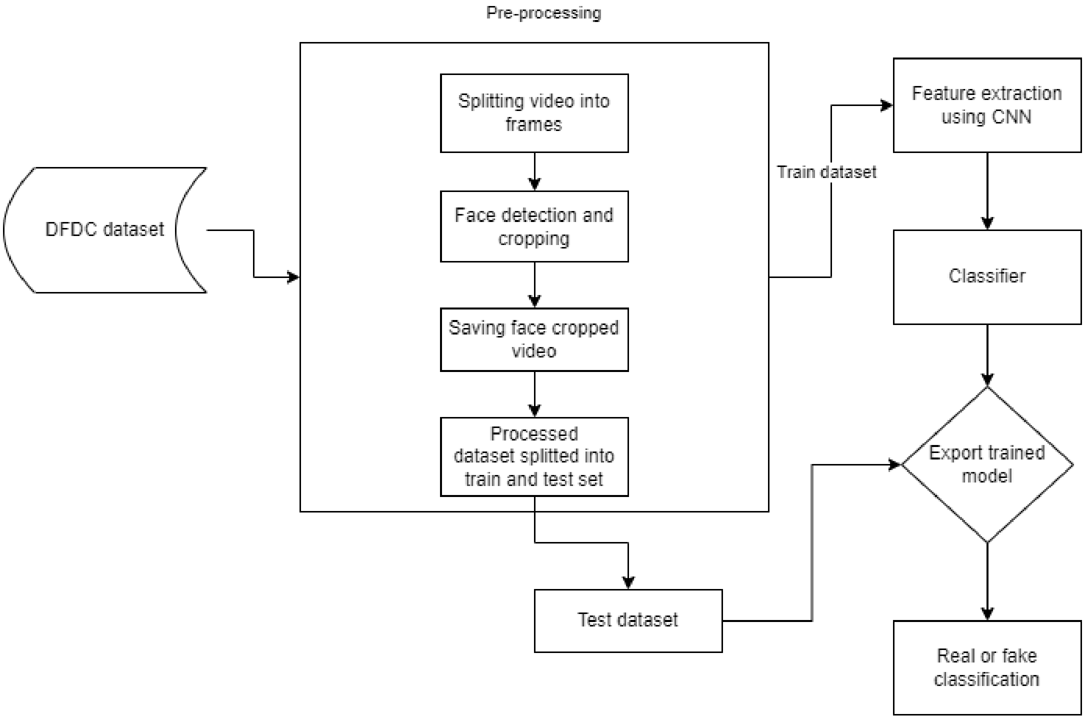

Pre-processing
DFDC dataset
Splitting video into frames
Face detection and cropping
Saving face cropped video
Processed dataset splitted into train and test set
Train dataset
Test dataset
Feature extraction using CNN
Classifier
Export trained model
Real or fake classification

Figure 3.1: Methodology flowchart

Temporal Sequence Modeling with LSTM: To capture temporal dependencies and patterns across the frames, the extracted ResNet features are fed into an LSTM network. LSTM is a type of recurrent neural network (RNN) that is designed to model and analyze sequences of data. By processing the ResNet features sequentially, the LSTM network can learn long-term dependencies and patterns in the video sequence.

LSTM Classifier: The LSTM network's output is passed through one or more fully connected layers to perform classification between real and deepfake videos. The fully connected layers learn to map the LSTM's output to the appropriate classification decision. The final layer typically utilizes a softmax activation function to provide probabilities for each class, indicating the likelihood of the input video being real or a deepfake.

Training and Evaluation: The ResNet-LSTM model is trained using a dataset that consists of labeled deepfake and real videos. During training, the model learns to extract features using the ResNet architecture and analyze the temporal sequence using the LSTM network. The model's weights are optimized by minimizing a suitable loss function, such as categorical cross-entropy. The trained model is then evaluated on a separate testing dataset to assess its performance in deepfake detection. Accuracy, precision, recall, and F1 score are among the most commonly used metrics for evaluation.

The goal of this method is to detect deepfakes by utilizing ResNet's feature extraction abilities along with LSTM's temporal modeling capabilities to capture both spatial and temporal cues. The ResNet-LSTM model can learn discriminative features from the frames of real and deepfake videos and effectively analyze the sequential information present in the video sequence to make accurate classification decisions.

It should be emphasized that the effectiveness of identifying deepfakes with ResNet and an LSTM classifier is reliant on the existence of comprehensive and inclusive training data. Additionally, hyperparameter tuning, regularization techniques, and other model optimization strategies can further enhance the model's performance. The field of deepfake detection is continuously evolving, and researchers are exploring various architectures and methodologies to improve the robustness and accuracy of deepfake detection models.

## 3.2 DATASET SPECIFICS

To train these deep learning models, large datasets of both real and deepfake videos are required. These datasets enable the models to learn the features and patterns specific to deepfakes and generalize their detection capabilities. Researchers are actively curating and expanding deepfake datasets to improve the accuracy and reliability of deepfake detection algorithms.

### 3.2.1 Celeb-DF

The Celeb-DF dataset is a popular and widely used dataset in the field of deepfake detection research. It was created specifically for the task of detecting deepfake videos, which are manipulated videos that use artificial intelligence techniques to alter the appearance or behavior of individuals.

The Celeb-DF dataset consists of videos extracted from the Internet and covers a wide range of celebrities. It contains both real and deepfake videos, making it suitable for training and evaluating deepfake detection models. The dataset was developed to address the growing concern over the potential misuse of deepfake technology for malicious purposes, such as spreading misinformation or creating non-consensual explicit content. Key characteristics of the Celeb-DF dataset:

Size and Diversity: The Celeb-DF dataset is relatively large, containing thousands of video clips from various celebrities. It includes a diverse set of individuals, encompassing different ages, genders, and ethnicities. This diversity is crucial for training deepfake detection models that can generalize well across different demographics.

Annotation and Labels: Each video clip in the Celeb-DF dataset is annotated to indicate whether it is a real video or a deepfake. This binary classification annotation allows researchers to train and evaluate deepfake detection algorithms accurately. The ground truth labels provided with the dataset are essential for benchmarking the performance of different detection models.

Video Quality and Variation: The Celeb-DF dataset includes videos of varying quality, resolution, and compression artifacts. This variation is crucial to simulate real-world conditions where deepfake videos can be of different visual quality due to various factors like source material, editing techniques, and compression algorithms.

Data Splits: The Celeb-DF dataset typically provides predefined data splits for training, validation, and testing. These splits ensure consistent evaluation and comparison of different deepfake detection models across different research works.

The Celeb-DF dataset is extensively utilized in the research field to create and assess deepfake detection algorithms. It functions as a reference dataset to gauge the effectiveness of various methods and techniques in differentiating between authentic and tampered videos. This dataset has propelled the advancement of deepfake detection technology and has played a vital role in developing dependable and resilient methods to counter the proliferation of misleading and deceitful content.

It's worth noting that the Celeb-DF dataset represents a specific subset of deepfake videos, focusing on celebrity faces. While it provides valuable insights and knowledge in the field, it may not fully represent the diversity and complexity of deepfake videos found in real-world scenarios. Therefore, it is crucial to continue expanding and refining datasets to encompass a broader range of contexts and applications for more robust and reliable deepfake detection systems.

### 3.2.2 Face forensics++

The FaceForensics++ dataset is a comprehensive and widely used benchmark dataset designed for advancing the research and development of deepfake detection algorithms. It was created to address the increasing concern over the proliferation of manipulated videos, particularly deepfake videos, which are generated using artificial intelligence techniques to alter facial expressions and appearances.

The FaceForensics++ dataset encompasses a diverse range of deepfake manipulation techniques and provides a large collection of videos to facilitate rigorous evaluation and comparison of detection methods. It includes both real and manipulated videos, covering various levels of sophistication and realism. The dataset was designed to capture the evolving landscape of deepfake techniques, allowing researchers to develop robust and effective detection algorithms. Key features of the FaceForensics++ dataset:

Manipulation Techniques: The dataset contains videos that are manipulated using different deepfake generation techniques, such as face swapping, face reenactment, and facial expression synthesis. This variety of manipulation techniques helps researchers understand and address the challenges associated with different types of deepfake videos.

Source Videos: The FaceForensics++ dataset includes source videos of individuals that are used as the basis for generating the manipulated videos. These source videos provide the real reference for comparison and analysis, enabling researchers to examine the characteristics of deepfake manipulations in relation to the original content.

Realism and Diversity: The dataset contains various manipulated videos with different levels of realism, providing a wide range of scenarios to evaluate the effectiveness of deepfake detection algorithms. These scenarios include low-quality deepfakes, high-quality deepfakes, and videos created using advanced AI techniques.

Ground Truth and Evaluation: Each video in the FaceForensics++ dataset is labeled to indicate whether it is real or manipulated. These ground truth labels are essential for training and evaluating deepfake detection models. The dataset provides predefined data splits for training, validation, and testing to ensure standardized evaluation procedures.

Large Scale: The FaceForensics++ dataset is significantly larger in scale compared to previous deepfake datasets. It consists of thousands of video sequences, providing ample data for training and evaluating deepfake detection algorithms and enabling robust statistical analysis.

The FaceForensics++ dataset has been crucial in the progression of deepfake detection, enabling the creation and testing of various advanced detection techniques. It is now a widely recognized benchmark for evaluating the efficacy of deepfake detection algorithms. Dataset continues to evolve, with ongoing efforts to expand its diversity, complexity, and coverage of emerging deepfake techniques.

It is important to note that the FaceForensics++ dataset represents a specific subset of deepfake manipulation techniques and may not capture the full range of possible manipulations encountered in real-world scenarios. Therefore, it is crucial to continually update and improve datasets to reflect the evolving landscape of deepfake technologies and address the challenges associated with detecting increasingly sophisticated manipulations.

## 3.3 DATA PREPARATION

The preprocessing stage is a crucial step in deepfake video detection as it involves preparing the data for input to the deep learning models. The goal of preprocessing is to enhance the quality of the data, extract relevant features, and normalize the input to improve the accuracy of the subsequent detection algorithms.

### 3.3.1 Preprocessing

Here are some common preprocessing steps in deepfake video detection:

Video frame extraction: Deepfake videos are typically composed of a sequence of frames. The first step is to extract individual frames from the video. These frames serve as the input data for subsequent analysis. Frame extraction can be performed using video processing libraries or tools that allow you to access each frame.

Face detection and alignment: Deepfake manipulation often focuses on facial regions, so accurate detection and alignment of faces are crucial. Face detection algorithms, such as Haar cascades or deep learning-based detectors like OpenCV's DNN module or MTCNN, can be employed to locate and extract faces from each frame. Once the faces are detected, they can be aligned to a standardized pose or cropped to remove unnecessary background information. This step ensures consistent and normalized input across frames.

Preprocessing of face regions: The extracted face regions can undergo additional preprocessing steps to improve the quality of the input data. These steps may include resizing the face images to a standard size, converting them to grayscale, or applying image enhancement techniques to enhance contrast or reduce noise. Normalization can be a part of preprocessing, which may involve subtracting the mean or dividing by the standard deviation to enable smoother convergence while training.

Data augmentation: To enhance the model's generalization capabilities and increase the training data's variety, data augmentation techniques can be utilized. Augmentation techniques include random rotations, translations, scaling, flips, and adding noise. These variations help the model learn to recognize deepfake artifacts under different conditions and increase its robustness to potential variations in real-world deepfake videos.

Deep learning models often require input in the form of feature vectors rather than raw images. Therefore, feature extraction methods can be applied to convert the preprocessed face images into meaningful representations. Techniques like deep neural networks or pre-trained models, such as VGGFace, ResNet, or Inception, can be utilized to extract high-level features from the face regions. These extracted features can capture important visual cues and patterns that are indicative of deepfake manipulation.

Dataset balancing: Balancing the dataset is essential to prevent bias in training the deep learning models. Deepfake videos are relatively scarce compared to real videos, so careful consideration should be given to ensure an equal representation of both classes in the training data. Techniques such as oversampling, undersampling, or using generative models can be employed to balance the dataset effectively.

By performing these preprocessing steps, the input data is refined and prepared for training or inference with deep learning models. The quality of the preprocessing stage significantly impacts the subsequent detection accuracy and the model's ability to differentiate between deepfake and real videos. It is important to adapt the preprocessing techniques based on the characteristics of the dataset and the specific requirements of the deep learning model being utilized.

To improve the model's ability to make predictions in real time. It acquired the information from many publicly accessible data-sets, including FaceForensic++ and Celeb-DF. Then, in order to do precise and immediate detection on various types of movies, combined the datasets had collected. It has taken into consideration 50% real and 50% fake videos in order to prevent the model's training bias. As this paper does not cover audio deepfakes, the DFDC dataset was limited to only include a small number of videos with altered audio. A Python script was used to preprocess the DFDC dataset and remove the videos with altered audio.

For the experiment, a total of 500 real and 500 fake videos were taken from the Celeb-DF dataset, and 1000 real and 1000 fake videos were taken from the FaceForensic++ dataset. The videos were preprocessed to eliminate unnecessary noise, and face detection and cropping were used to keep the necessary length. The videos were divided into frames, and frames without faces were ignored during preprocessing.

A threshold value was selected based on the average of all the frames in each video to maintain the same number of frames, considering the capacity restrictions of computers. In the experimental environment, processing 300 frames at once was computationally challenging, so a threshold value of 150 frames was selected based on GPU processing capabilities. Only the first 150 frames of each video were saved to the new video, and the remaining frames were saved to the new dataset.

In this phase, the facial region is cropped out of the video frames that were in the frame.The chosen frames are then submitted to an MT-CNN based face detection algorithm to find any faces in the frame. It utilises this network because it performs well with faces of different sizes in the image and has a flexible design that uses a multi-task learning mechanism.

In this experiment it has been merely taken into account the first 150 frames sequentially to demonstrate how LSTM should be used. A frame rate of 30 frames per second and a resolution of 112 x 112 or (244 x 244) are saved for the newly created video. The Figure 3.2 shows resultant frames of the cropped videos with only blue colour pixel from the red-blue-green pixel.

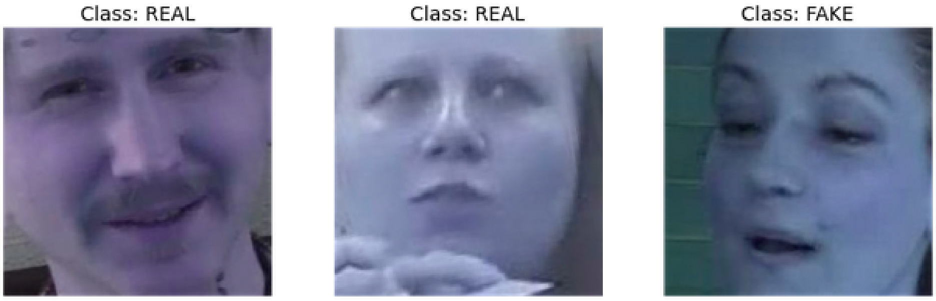


Figure 3.2: Processed real & fake frame of 3 different videos

## 3.4 FEATURE EXTRACTION

Deep learning models often require input in the form of feature vectors rather than raw images. Therefore, feature extraction methods can be applied to convert the preprocessed face images into meaningful representations. Techniques like deep neural networks or pre-trained models, such as VGGFace, ResNet, or Inception, can be utilized to extract high-level features from the face regions. These extracted features can capture important visual cues and patterns that are indicative of deepfake manipulation.

The feature extraction phase in deepfake video detection involves extracting meaningful and discriminative features from video frames or sequences. Here's an explanation of the feature extraction process:

Frame-level Feature Extraction: The video is divided into individual frames, typically at a fixed frame rate. Each frame represents a single image. Various image processing techniques can be applied to enhance the frames, such as resizing, normalization, or applying filters to improve consistency and quality.

Pretrained Convolutional Neural Networks (CNNs): Pretrained CNN models, such as VGGNet, ResNet, or InceptionNet, are commonly used for feature extraction. These models are trained on large-scale image datasets like ImageNet, enabling them to learn generic visual features that can be relevant for deepfake detection. The input frames are passed through the CNN layers, and the output activations from one or more intermediate layers are used as features.

Fine-tuning or Feature Extraction Only: Depending on the approach, the pretrained CNN can be used in two ways. In fine-tuning, the entire CNN model is further trained on a deepfake-specific dataset to adapt it to the detection task. Alternatively, the pretrained CNN can be used as a fixed feature extractor, where only the learned features from intermediate layers are used without further training.

Frame-level Feature Aggregation: The extracted features from individual frames are aggregated to create a representation for the entire video. Various aggregation methods can be employed, such as averaging, max pooling, or concatenation, to summarize the frame-level features into a fixed-length representation.

Temporal Modeling (Optional): In addition to frame-level feature extraction, temporal modeling techniques can be applied to capture the temporal dynamics and dependencies within the video sequence. Recurrent Neural Networks (RNNs), such as LSTMs or GRUs, can be employed to model the sequence of frame-level features and capture the temporal context.

Dimensionality Reduction (Optional): If the extracted features have high dimensionality, dimensionality reduction techniques, such as Principal Component Analysis (PCA) or t-SNE, can be applied to reduce the feature space while preserving the most relevant information.

The goal of the feature extraction phase is to transform the raw video frames into a compact and informative representation that captures the relevant visual information for distinguishing between real and deepfake videos. These extracted features serve as input to subsequent classification algorithms or models for deepfake detection.

It's important to note that the choice of CNN architecture, fine-tuning strategy, aggregation method, and temporal modeling technique can vary based on the specific deepfake detection approach and the available resources. Different combinations of these techniques can yield varying levels of accuracy and robustness in detecting deepfake videos.

### 3.4.1 Backbone CNN model

**XceptionNet**

Xception-Net, short for Extreme Inception, is a deep convolutional neural network architecture that was proposed by François Chollet in 2016. It is inspired by the Inception architecture, which was developed by Google for the ILSVRC.

The main idea behind Xception-Net is to rethink the way convolutions are performed in deep neural networks. Traditional convolutional layers use a small kernel size (e.g., 3x3) to capture local patterns in an input image. In contrast, Xception-Net separates the process of learning spatial relationships and channel relationships. It replaces the standard convolutional layers with depthwise separable convolutions.

Depthwise separable convolutions involve two distinct phases: depthwise convolutions and pointwise convolutions. During the depthwise convolution stage, individual filters are used to convolve each input channel, which captures spatial data. In the pointwise convolution stage, the output channels of the depthwise convolutions are combined using 1x1 convolutions, which captures inter-channel relationships.

The advantage of using depthwise separable convolutions is that they significantly reduce the computational cost compared to traditional convolutions while still maintaining good accuracy. This reduction in computational cost makes Xception-Net more efficient and allows it to be trained on large-scale datasets.

Xception-Net has demonstrated its effectiveness in various computer vision tasks, such as image classification, object detection, and semantic segmentation, and achieved state-of-the-art performance on the ImageNet dataset. It has become a popular architecture in the field of deep learning and has served as a backbone for many subsequent models.

### EfficientNet

EfficientNet is another deep neural network architecture that was proposed by Mingxing Tan and Quoc V. Le in 2019. It aims to achieve state-of-the-art performance in terms of both accuracy and efficiency by scaling the model in a principled manner. The key idea behind EfficientNet is to balance three different scaling dimensions: depth, width, and resolution. Traditional approaches typically focus on scaling only one of these dimensions, but EfficientNet introduces a compound scaling method that uniformly scales all three.

EfficientNet uses a baseline architecture similar to the MobileNetV2, which consists of depthwise separable convolutions. The compound scaling is achieved by systematically scaling the network width, depth, and resolution using a coefficient called "phi." The phi coefficient controls how much to scale each dimension.

By scaling the network width, depth, and resolution, EfficientNet achieves better trade-offs between accuracy and efficiency. It can be scaled up to obtain higher accuracy or scaled down to reduce model size and computational cost while maintaining competitive performance.

EfficientNet models are trained using a combination of techniques, including advanced data augmentation, AutoML, and neural architecture search. The neural architecture search helps to automatically discover the optimal network structure within the defined scaling constraints.

Overall, EfficientNet has become a popular architecture in the field of deep learning due to its ability to achieve excellent performance with efficient resource utilization.

### MobileNet

The MobileNet architecture is tailored for mobile and embedded devices that have restricted computational resources, as it is a lightweight convolutional neural network. It was introduced by Howard et al. in the paper "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications" in 2017.

The primary goal of MobileNet is to provide high accuracy in computer vision tasks while minimizing the number of parameters and computational complexity. It achieves this by utilizing depthwise separable convolutions and a few additional optimizations.

Key features of MobileNet:

- Depthwise Separable Convolutions: MobileNet replaces the standard convolutional layers with depthwise separable convolutions. A depthwise convolution operates by utilizing a singular convolutional filter per input channel, whereas a pointwise convolution conducts a 1x1 convolution to merge the output channels that come from the depthwise convolution. This decomposition effectively decreases the computational expense while maintaining the flow of information.

- Width Multiplier: MobileNet introduces a hyperparameter called the width multiplier (denoted by ) to control the model's size and computational complexity. It scales the number of input and output channels in each layer by a factor , effectively reducing the number of parameters and computations. A smaller  value results in a more compact and computationally efficient network.

- Resolution Multiplier: Another hyperparameter in MobileNet is the resolution multiplier (denoted by ), which scales down the spatial resolution of input images. It reduces the input dimensions, allowing for faster inference and lower memory requirements. However, reducing the resolution may also lead to some loss in accuracy.

- Additional Optimizations: MobileNet incorporates several optimizations to further improve its efficiency. These include the use of ReLU6 activation function, which clips the output values to the range [0, 6] instead of the usual [0, 1] range, and the addition of a global average pooling layer at the end of the network to reduce the spatial dimensions.

Benefits and Applications:

- Efficiency: MobileNet is designed to be highly efficient and lightweight, making it suitable for deployment on mobile devices, embedded systems, and edge devices. Its reduced computational complexity allows for real-time inference and low-latency applications.

- Model Size: MobileNet significantly reduces the number of parameters compared to traditional convolutional neural networks. This leads to smaller model sizes, making it easier to deploy and store on resource-constrained devices.

- Real-time Applications: Due to its computational efficiency, MobileNet is well-suited for real-time computer vision applications, such as object detection, image classification, facial recognition, and augmented reality on mobile and embedded platforms.

- Transfer Learning: MobileNet can be used as a base network for transfer learning. Pretrained MobileNet models trained on large-scale datasets, such as ImageNet, can be fine-tuned on specific tasks with limited labeled data, saving training time and resources.

MobileNet has become a popular choice for mobile and embedded vision applications, enabling deep learning capabilities on devices with limited computational power and memory constraints. Its efficiency, compact size, and ability to achieve a good balance between accuracy and resource usage make it a valuable tool in the field of computer vision.

### ShuffleNet

ShuffleNet is a convolutional neural network architecture designed to achieve a good trade-off between accuracy and computational efficiency, specifically targeting applications with limited computational resources such as mobile and embedded devices. It was introduced in the paper ”ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices” by Xiangyu Zhang et al. in 2018.

The main idea behind ShuffleNet is to reduce the computational complexity of the network by utilizing group convolutions and channel shuffling operations.

The architecture consists of a series of building blocks called Shuffle units, which are responsible for shuffling and recombining feature maps across channels.

Key features of ShuffleNet:

- Channel Shuffle: The Channel Shuffle operation is the core component of ShuffleNet. It enables information exchange between different groups of channels, allowing the network to learn richer representations. By introducing this operation, ShuffleNet reduces the computational cost associated with cross-channel communication.

- Group Convolutions: ShuffleNet divides the channels into groups and performs separate convolutions on each group. This reduces the number of parameters and computations required compared to traditional convolutional networks, where all channels are processed together.

- Bottleneck Structure: ShuffleNet employs a bottleneck structure similar to other modern network architectures, which consists of a pointwise convolution (1x1 filter), a channel shuffle operation, and a depthwise convolution (3x3 filter). This structure enables efficient information flow while reducing the computational cost.

- Different ShuffleNet Versions: ShuffleNet has multiple versions, such as ShuffleNet v1, ShuffleNet v2, and ShuffleNet v2 with improved accuracy. These versions vary in terms of network depth, number of channels, and overall computational complexity. ShuffleNet v2, in particular, achieves higher accuracy by employing a combination of pointwise and depthwise separable convolutions.

ShuffleNet has gained popularity for its ability to strike a balance between accuracy and efficiency. Its innovative design, incorporating channel shuffling and group convolutions, has opened up new possibilities for deploying deep neural networks on resource-constrained devices without sacrificing performance.

## 3.5 LEARNING FEATURES & CLASSIFICATION

In a variety of contexts, including challenging computer vision tasks, machine translation, face recognition, object identification, and localization, learned features have demonstrated their effectiveness in resolving complex problems. Deep learning techniques use a black box CNN feature extraction procedure to automatically learn and derive features from the training data. Four primary phases make up the deep learning-based methodology's basic structure: a. Data preparation, b. Extraction of features, c. Learning features and Classification.

Deep learning is a subfield of machine learning that focuses on training artificial neural networks to recognize patterns and make predictions. It has proven to be effective in various computer vision tasks, including image and video analysis, which makes it a suitable approach for deepfake detection.

The following are some common approaches for deepfake video detection that leverage deep learning techniques:

Convolutional Neural Networks: CNNs are widely used for image and video analysis tasks. They excel at extracting visual features from input data and can be employed to identify anomalies or artifacts specific to deepfake videos. CNN-based models can learn to differentiate between real and manipulated frames by learning the underlying patterns and inconsistencies introduced by deepfake algorithms.

Recurrent Neural Networks: RNNs, particularly Long Short-Term Memory (LSTM) networks, are effective in analyzing sequential data such as video frames. These networks can learn temporal dependencies and identify irregularities in the flow of frames caused by deepfake manipulation. By considering the contextual information across multiple frames, RNNs enhance the accuracy of deepfake detection models.

Capsule Networks: Capsule Networks are an emerging type of neural network architecture that aims to improve the robustness of CNNs. They focus on capturing hierarchical relationships between visual elements in an image or video. By utilizing capsule networks, deepfake detection models can potentially detect subtle visual inconsistencies and discrepancies in facial features that are indicative of manipulation.

Generative Adversarial Networks: GANs are often used to generate deepfake videos, but they can also be utilized in their detection. By training a GAN to distinguish between real and fake videos, it is possible to develop a discriminator network that becomes proficient at identifying deepfake content. This approach leverages the adversarial relationship between the generator and the discriminator to enhance the detection accuracy.

Motion Analysis: Deepfake videos often exhibit unnatural or irregular motion patterns due to the manipulation process. Deep learning algorithms can be applied to analyze and compare the motion characteristics of real and fake videos. By focusing on motion cues, such as optical flow or skeletal joint movements, deep learning models can identify discrepancies that are indicative of deepfake content.

To train these deep learning models, large datasets of both real and deepfake videos are required. These datasets enable the models to learn the features and patterns specific to deepfakes and generalize their detection capabilities. Researchers are actively curating and expanding deepfake datasets to improve the accuracy and reliability of deepfake detection algorithms.

The classification phase in deepfake video detection using Long Short-Term Memory involves utilizing the extracted features from the video frames and feeding them into an LSTM network to make predictions. Here's an explanation of the classification process using LSTM:

Input Preparation: The extracted features from the video frames, obtained during the feature extraction phase, are organized into a sequence. Each frame's features are treated as a time step in the sequence.

LSTM Network Architecture: The LSTM network is designed to analyze sequential data and capture temporal dependencies. It consists of LSTM layers, which have memory cells and gates to retain and propagate information over time. The number of LSTM layers and their sizes can vary based on the complexity of the task and available resources.

Sequence Modeling: The LSTM network takes the sequential features as input and processes them through the network's layers. At each time step, the LSTM layer updates its internal state and makes predictions based on the current input and the information it has accumulated from previous time steps. This allows the network to capture temporal patterns and dependencies present in the video sequence.

Classification Layer: After processing the entire sequence, the output of the last LSTM layer is typically fed into one or more fully connected layers. These layers learn to map the LSTM's output to the appropriate class labels, such as real or fake. The final layer often employs a softmax activation function to produce probabilities for each class, indicating the likelihood of the input video being real or a deepfake.

Training and Evaluation: The LSTM network is trained using labeled training data, where each video sequence is associated with the correct class label. The network's weights are optimized by minimizing a suitable loss function, such as categorical cross-entropy, through backpropagation and gradient descent. The trained LSTM model is then evaluated on a separate testing dataset to assess its performance in classifying deepfake videos. Common evaluation metrics include accuracy, precision, recall, and F1 score.

The LSTM network's ability to capture long-term dependencies and patterns in the temporal sequence of features makes it suitable for deepfake video classification. By analyzing the sequential information, the LSTM model can learn to distinguish between real and manipulated videos based on the extracted features.

It's important to note that the performance of deepfake detection using LSTM depends on factors such as the quality and diversity of the training data, the architecture and hyperparameter settings of the LSTM network, and the availability of computational resources for training and inference. Researchers continuously explore variations of LSTM architectures, such as bidirectional LSTMs or attention mechanisms, to improve the accuracy and robustness of deepfake detection models.

# Chapter 4

# RESULTS AND ANALYSIS

This chapter consists of experimental results of deep-fake video detection using EfficientNet architecture. The EfficientNet family consists of 8 models and all these models were trained. Hence the best performing EfficientNet is being further attached along with LSTM layer as per the architecture of the base paper. Further it also focuses on the comparitive analysis with different backbone CNN architecture for better understanding.

## 4.1 EVALUATION METRICS

Standard evaluation metrics used for assessing the DL models' performance for deepfake detection. The metrics are accuracy, Precision, Recall, and F-score, defined concerning TP, TN, FP, and FN parameters. True positive (TP) measures the number of times the predicted result is a fake video, with the actual result also being a fake video. True Negative (TN) provides the number of times the predicted result is a real video, with the actual result also being a real video. False positive (FP) gives the number of times the predicted result is a real video, with the actual result being a fake video. False negative (FN) gives the number of times the predicted result is a fake video with the actual result being a real video.

Based on these values, accuracy, precision, and recall are calculated. Accuracy is the fraction of the correct model predictions to the total model predictions. In equation. 4.1 it is given by the ratio of sum of true positives and true negatives to the sum of true positives, true negatives, false positives and false negatives. Precision is the ratio of true positives to the sum of true positives and false positives, it is given by the equation 4.2. Recall is the ratio of true positives to the sum of true positives and false negatives, given by equation 4.3 and finally, the F1-Score is computed as the harmonic mean of precision and recall given by the equation 4.4. For the evaluation of the classification task, used accuracy, precision, recall, and f1-score had the most used evaluation metrics in case of an imbalanced dataset F1-score has become a more important metric;

otherwise, all metrics will give equal importance.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

$$F - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.4}$$

Table 4.1: Accuracy of different EfficientNet architecture

| **Model** | **Accuracy %** |
|---|---|
| EfficientNet-B0 | 84.61 |
| EfficientNet-B1 | 86.32 |
| EfficientNet-B2 | 85.47 |
| EfficientNet-B3 | 82.9 |
| EfficientNet-B4 | 71.79 |

Table 4.1 presents the observations with reference to the performance of the different EfficientNet architectures when trained on Celeb-DF dataset. And it shows – architecture ensures better performance in terms of no of parameters trained and accuracy. Hence it is being further used along with LSTM architecture for classification over both dataset namely Face-Forensics++ and Celeb-DF.

Table 4.2: Results for Celeb-DF dataset

| Model | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| ResNet + LSTM | 89.74 | 0.867 | 0.915 | 0.89 |
| EfficientNet+LSTM | 84.61 | 0.913 | 0.801 | 0.853 |
| EfficientNet | 72.22 | 0.704 | 0.723 | 0.713 |
| XceptionNet + LSTM | 82.48 | 0.79 | 0.848 | 0.818 |
| ShuffleNet + LSTM | 72.22 | 0.823 | 0.69 | 0.751 |
| DenseNet + LSTM | 85.04 | 0.974 | 0.783 | 0.868 |
| MobileNet + LSTM | 88.46 | 0.93 | 0.853 | 0.889 |

The Table 4.2 shows the performance metrics (Accuracy, Precision, Recall, F1-score) for different models that combine convolutional neural network (CNN) architectures with a long short-term memory (LSTM) layer. These models are likely used for some kind of classification task. Let's break down the metrics:

Accuracy: It represents the overall correctness of the model's predictions and is calculated as the ratio of correct predictions to the total number of predictions. Figure 4.1 shows accuracy variation between different models for celeb-df dataset. For example, the ResNet + LSTM model achieves an accuracy of 89.74%, indicating that it correctly classifies 89.74% of the samples.

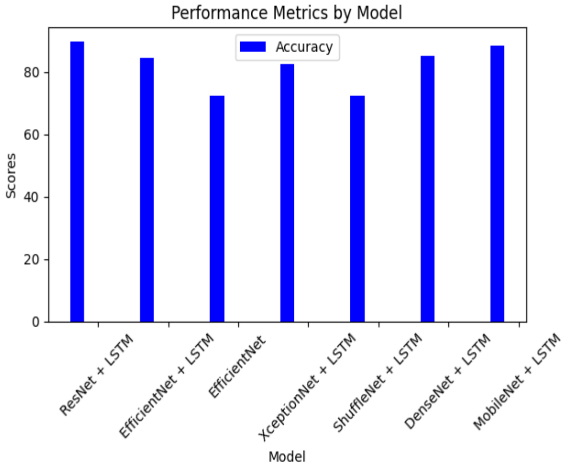


Figure 4.1: Bar graph representing accuracy for different models

Precision: Precision measures the proportion of correctly predicted positive samples out of the total predicted positive samples. It is a measure of the model's ability to avoid false positives. Higher precision indicates fewer false positives. Figure 4.2 shows precision, recall, F1-score variation between different models for celeb-df dataset. For instance, the DenseNet + LSTM model achieves a precision of 0.974, indicating it has a high proportion of correct positive predictions.

Recall: Recall, also known as sensitivity or true positive rate, measures the proportion of correctly predicted positive samples out of the total actual positive samples. It is a measure of the model's ability to find all the positive samples. Higher recall indicates fewer false negatives. For example, the EfficientNet + LSTM model achieves a recall of 0.801, meaning it captures 80.1% of the actual positive samples.

F1-score: The F1-score is the harmonic mean of precision and recall, providing a balanced measure of a model's performance. It takes into account both

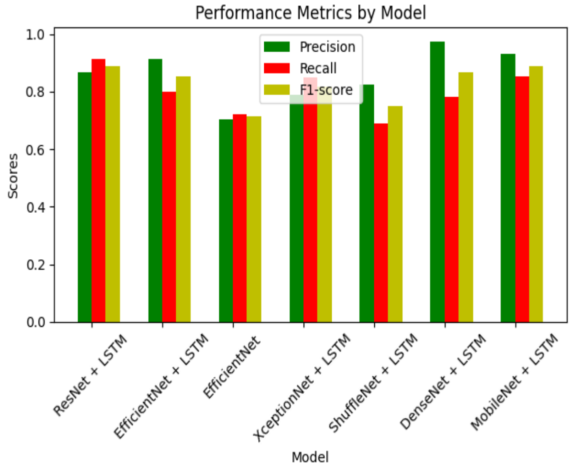


Figure 4.2: Graph representing precision, recall, F1 score for different models

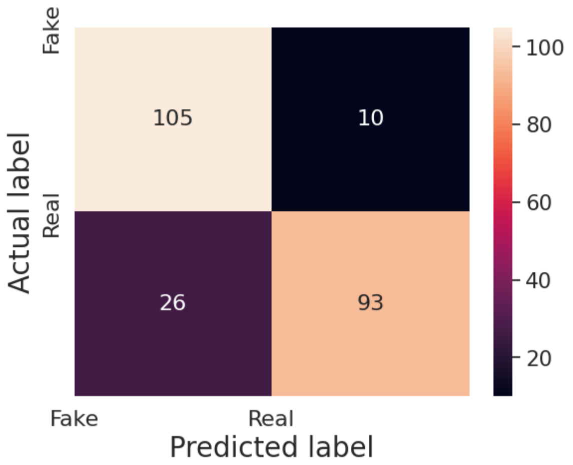


Figure 4.3: Confusion matrix for EfficientNet + LSTM model on Celeb-DF dataset

false positives and false negatives. The F1-score reaches its best value at 1 and worst value at 0. For instance, the ResNet + LSTM model achieves an F1-score of 0.89, indicating a relatively balanced performance between precision and recall.

Figure 4.3 shows confusion matrix of EfficientNet + LSTM model for celeb-df dataset. Based on the provided metrics, the models exhibit varying levels of performance. The ResNet + LSTM model achieves the highest accuracy (89.74%) and F1-score (0.89), suggesting it performs well overall. The DenseNet + LSTM model shows high precision (0.974), indicating a low rate of false positives. The EfficientNet + LSTM model achieves relatively balanced precision (0.913) and

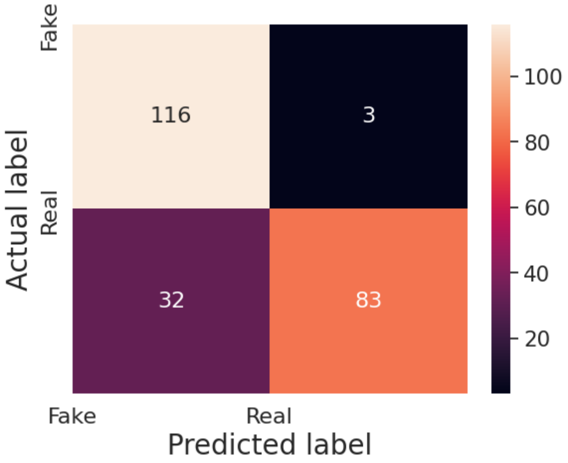


Figure 4.4: Confusion matrix for DenseNet + LSTM model on Celeb-DF dataset

recall (0.801) but has a lower accuracy (84.61%) compared to ResNet + LSTM. Figure 4.4 shows confusion matrix of XceptionNet + LSTM model for celeb-df dataset.

It's important to note that the choice of the best model depends on the specific requirements and priorities of the task at hand. Different metrics may be more important in different scenarios.

Table 4.3: Results for Face-Forensics++ dataset

| Model | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| XceptionNet + LSTM | 88.44 | 0.887 | 0.878 | 0.882 |
| MobileNet + LSTM | 85.67 | 0.76 | 0.937 | 0.839 |
| ShuffleNet + LSTM | 75.63 | 0.777 | 0.743 | 0.759 |
| DenseNet + LSTM | 94.22 | 0.939 | 0.944 | 0.941 |
| EfficientNet + LSTM | 90.45 | 0.893 | 0.912 | 0.902 |
| EfficientNet | 89.69 | 0.931 | 0.879 | 0.904 |
| ResNet + LSTM | 90.2 | 0.921 | 0.879 | 0.899 |

The Table 4.3 presents the performance metrics (Accuracy, Precision, Recall, F1-score) for different models that combine various convolutional neural network (CNN) architectures with a long short-term memory (LSTM) layer. These models are likely used for a classification task. Let's analyze the metrics:

Accuracy: Accuracy represents the overall correctness of the model's predictions and is calculated as the ratio of correct predictions to the total number of predictions. Figure 4.5 shows accuracy variation between different models for face forensics++ dataset. In this case, the DenseNet + LSTM model achieves the highest accuracy of 94.22%, indicating that it correctly classifies 94.22% of the samples.

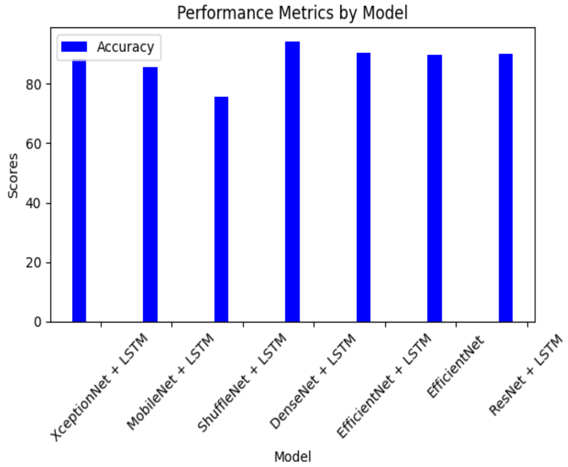


Figure 4.5: Bar graph representing accuracy for different models

Precision: Precision measures the proportion of correctly predicted positive samples out of the total predicted positive samples. Figure 4.6 shows precision, recall, F1-score variation between different models for face forensics++ dataset. It is a measure of the model's ability to avoid false positives. The DenseNet + LSTM model achieves a precision of 0.939, indicating a high proportion of correct positive predictions.

Recall: Recall, also known as sensitivity or true positive rate, measures the proportion of correctly predicted positive samples out of the total actual positive samples. It is a measure of the model's ability to find all the positive samples. The MobileNet + LSTM model achieves the highest recall of 0.937, suggesting it captures 93.7% of the actual positive samples.

F1-score: The F1-score is the harmonic mean of precision and recall, providing a balanced measure of a model's performance. It takes into account both false positives and false negatives. The DenseNet + LSTM model achieves the highest F1-score of 0.941, indicating a balanced performance between precision and recall.

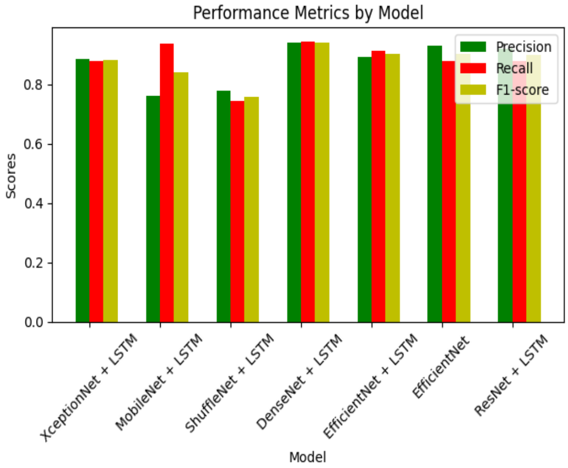


Figure 4.6: Graph representing precision, recall, F1 score for different models

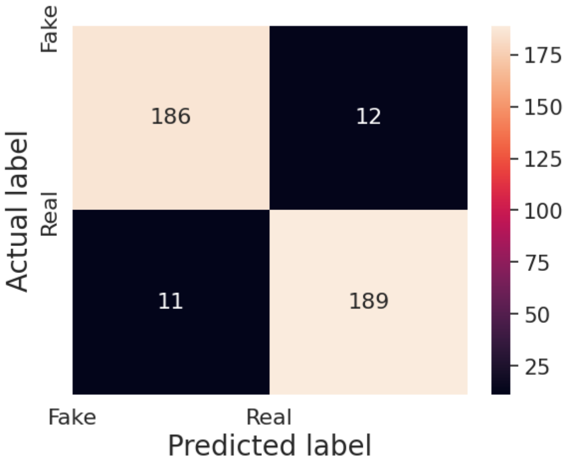


Figure 4.7: Confusion matrix for DenseNet + LSTM model on FaceForensics++ dataset

Figure 4.7 shows confusion matrix of DenseNet + LSTM model for face forensics dataset. Based on the provided metrics, the DenseNet + LSTM model demonstrates the highest performance across accuracy, precision, recall, and F1-score. It achieves high accuracy and F1-score, indicating a well-rounded performance. The MobileNet + LSTM model exhibits high recall, indicating its ability to capture a high proportion of positive samples. The XceptionNet + LSTM and EfficientNet + LSTM models show competitive performance with relatively balanced precision, recall, and F1-score. Figure 4.8 shows confusion matrix of EfficientNet + LSTM model for face forensics dataset.

It's important to consider the specific requirements and priorities of the task

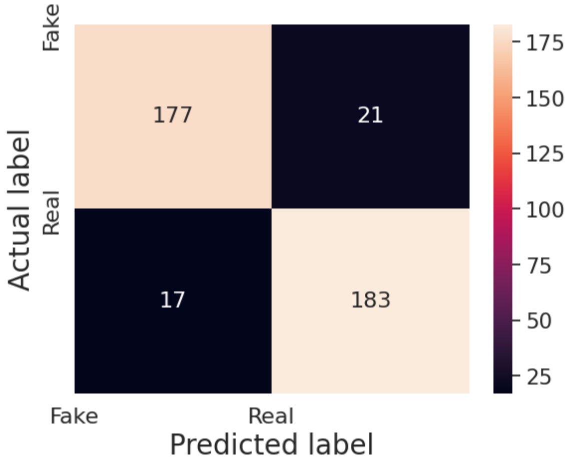


Figure 4.8: Confusion matrix for EfficientNet + LSTM model on FaceForensics++ dataset

when selecting the best model. Different metrics may be more important depending on the application and the trade-offs between precision and recall that are acceptable.

It can be observed from the measurements that the models' performance differs between datasets. With the highest accuracy and F1-score on both datasets, DenseNet + LSTM consistently performs well. Additionally, MobileNet + LSTM performs admirably, particularly in terms of recall. The task's precise needs and the trade-offs between various performance measures determine which model is the best to use.

# Chapter 5

# CONCLUSION AND FUTURE WORK

In conclusion, deep learning models have shown promising performance in the field of deepfake detection, which aims to identify manipulated or synthetic media content, such as images or videos. Several deep learning architectures have been applied to this task, including ResNet, EfficientNet, XceptionNet, MobileNet, ShuffleNet, and DenseNet, often in combination with recurrent neural networks like LSTM.

The performance of these models can vary depending on the specific dataset and evaluation metrics used. However, some general observations can be made:

DenseNet-based models consistently demonstrate high performance across datasets, achieving high accuracy, precision, recall, and F1-score. They are often a reliable choice for deepfake detection tasks.

MobileNet-based models tend to have high recall, making them effective at capturing a high proportion of actual positive samples. However, they might have slightly lower precision compared to other models.

XceptionNet-based models generally show competitive performance with balanced precision, recall, and F1-score.

EfficientNet-based models exhibit a trade-off between precision and recall, with varying performance depending on the specific dataset.

It's worth noting that the performance of deep learning models for deepfake detection is continuously evolving as new architectures, techniques, and datasets are introduced. Additionally, the choice of the best model depends on the specific requirements, available computational resources, and the trade-offs between precision, recall, and other evaluation metrics. Therefore, it is recommended to carefully evaluate and select the appropriate model based on the specific needs of the application.

From the metrics, it can be observed that the performance of the models varies across the datasets. DenseNet + LSTM consistently achieves high performance across both datasets, with the highest accuracy and F1-score. MobileNet + LSTM also demonstrates competitive performance, especially in terms of recall. The choice of the best model depends on the specific requirements of the task and the trade-offs between different performance metrics.

# REFERENCES

Afchar, D., Nozick, V., Yamagishi, J. & Echizen, I. (2018), Mesonet: a compact facial video forgery detection network, pp. 1–7.

Awotunde, J. B., Jimoh, R. G., Imoize, A. L., Abdulrazaq, A. T., Li, C.-T. & Lee, C.-C. (2023), 'An enhanced deep learning-based deepfake video detection and classification system', *Electronics* **12**(1).
**URL:** *https://www.mdpi.com/2079-9292/12/1/87*

Ciftci, U. A. & Demir, I. (2019), 'Fakecatcher: Detection of synthetic portrait videos using biological signals', *IEEE transactions on pattern analysis and machine intelligence* **PP**.

*DeFaking Deepfakes: Understanding Journalists' Needs for Deepfake Detection* (2020), USENIX Association.

Girgis, S., Amer, E. & Gadallah, M. E. (2018), 'Deep learning algorithms for detecting fake news in online text', *2018 13th International Conference on Computer Engineering and Systems (ICCES)* pp. 93–97.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2020), 'Generative adversarial networks', *Commun. ACM* **63**(11), 139–144.
**URL:** *https://doi.org/10.1145/3422622*

Güera, D. & Delp, E. J. (2018), Deepfake video detection using recurrent neural networks, *in* '2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)', pp. 1–6.

Heo, Y.-J., Choi, Y.-J., Lee, Y.-W. & Kim, B.-G. (2021), 'Deepfake Detection Scheme Based on Vision Transformer and Distillation', *arXiv e-prints* p. arXiv:2104.01353.

Laptev, I., Marszalek, M., Schmid, C. & Rozenfeld, B. (2008), Learning realistic human actions from movies.

Li, Y., Chang, M.-C., Farid, H. & Lyu, S. (2018), 'In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking'.

Li, Y., Chang, M.-C. & Lyu, S. (2018), In ictu oculi: Exposing ai created fake videos by detecting eye blinking, pp. 1–7.

Li, Y. & Lyu, S. (2018), 'Exposing deepfake videos by detecting face warping artifacts'.

Lu, C., Liu, B., Zhou, W., Chu, Q. & Yu, N. (2021), Deepfake video detection using 3d-attentional inception convolutional neural network, pp. 3572–3576.

Montserrat, D., Hao, H., Yarlagadda, S., Baireddy, S., Shao, R., Horvath, J., Bartusiak, E., Yang, J., Guera, D., Zhu, F. & Delp, E. (2020), Deepfakes detection with automatic face weighting, pp. 2851–2859.

Nguyen, H., Fang, F., Yamagishi, J. & Echizen, I. (2019), Multi-task learning for detecting and segmenting manipulated facial images and videos, pp. 1–8.

Nguyen, H., Yamagishi, J. & Echizen, I. (2019), Capsule-forensics: Using capsule networks to detect forged images and videos, pp. 2307–2311.

Tolosana, R., Vera-Rodríguez, R., Fierrez, J., Morales, A. & Ortega-Garcia, J. (2020), 'Deepfakes and beyond: A survey of face manipulation and fake detection', *Inf. Fusion* **64**, 131–148.

Tran, V.-N., Lee, S.-H., Le, H.-S. & Kwon, K.-R. (2021), 'High performance deepfake video detection on cnn-based with attention target-specific regions and manual distillation extraction', *Applied Sciences* **11**(16).
**URL:** *https://www.mdpi.com/2076-3417/11/16/7678*

Zhang, W. & Zhao, C. (2020), 'Exposing face-swap images based on deep learning and ela detection', *Proceedings* **46**(1).
**URL:** *https://www.mdpi.com/2504-3900/46/1/29*

Zhou, P., Han, X., Morariu, V. & Davis, L. (2017), Two-stream neural networks for tampered face detection, pp. 1831–1839.
0DĐorđević et al.

Đorđević, M., Milivojević, M. & Gavrovska, A. (2020), Deepfake video production and sift-based analysis.

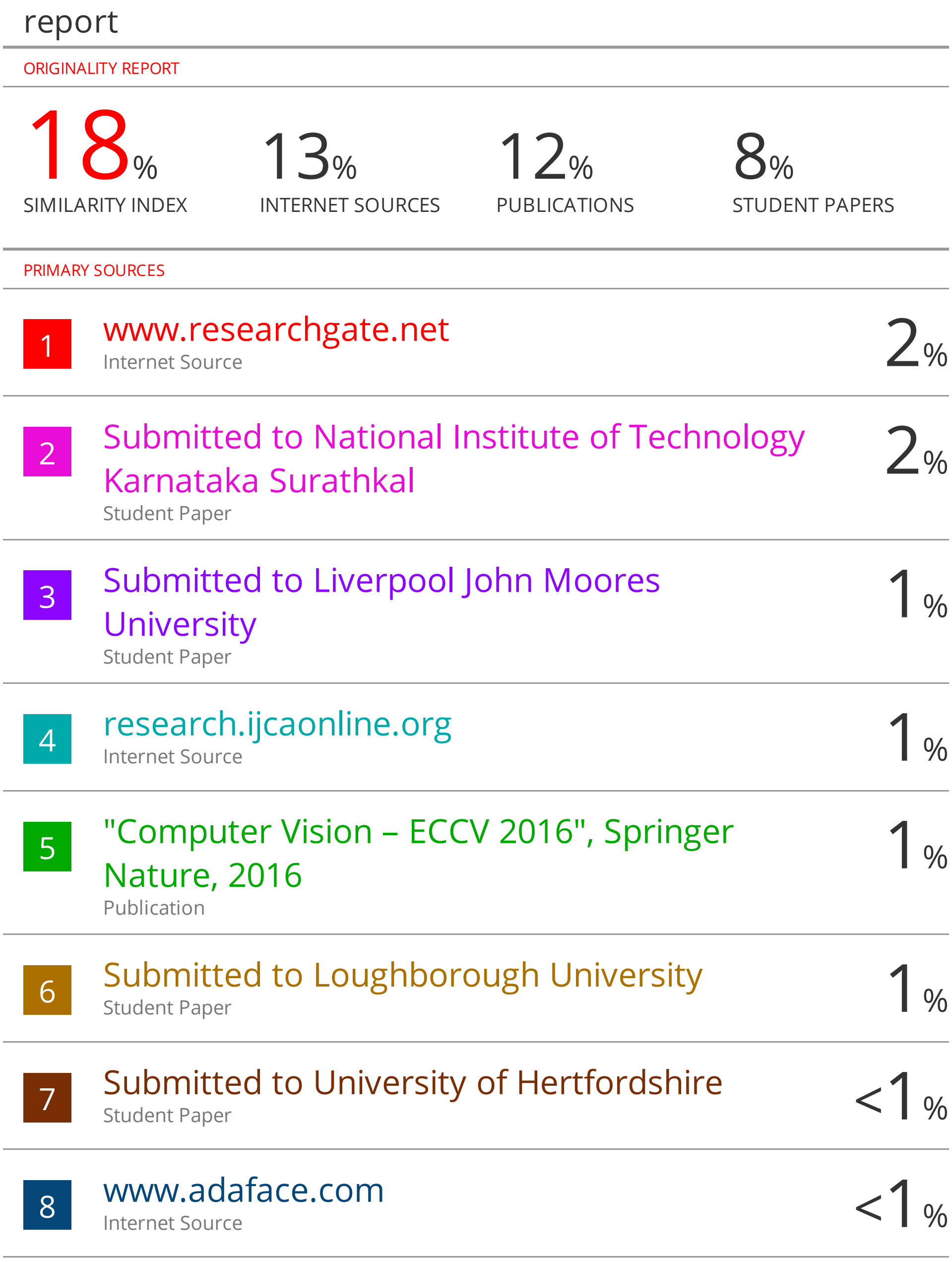
# Appendix A

# BIO-DATA

| | |
|---|---|
| **Name:** | Sourabh Makhija |
| **Address:** | R.S. Mamta Villa,<br>Jal Vihar Colony,Near old karmachari office,<br>Telibandha(Post),<br>Raipur-492001,Chhattisgarh(State) |
| **Email:** | makhijasourabh22@gmail.com |
| **Contact No.:** | +91 9575903054 |
| **Qualification:** | B.E. in Information Technology,<br>Shri Shankaracharya Institute of Professional Management and Technology Raipur, Chhattisgarh<br>M.Tech in Information Technology ,<br>National Institute of Technology Karnataka Surathkal |

# Appendix B

# TURNITIN CHECK



report

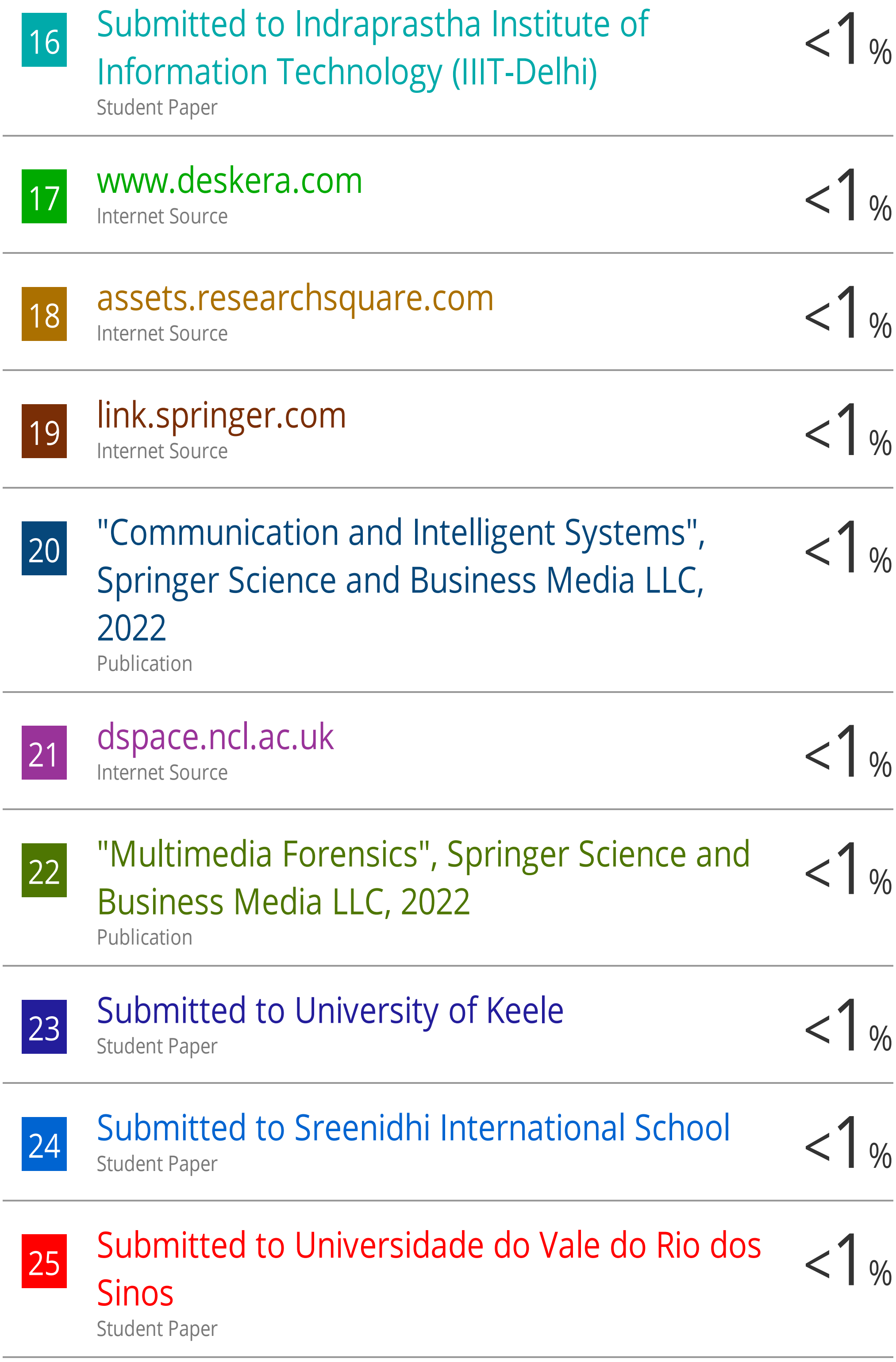
ORIGINALITY REPORT

| 18% | 13% | 12% | 8% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | www.researchgate.net<br>Internet Source | 2% |
| 2 | Submitted to National Institute of Technology Karnataka Surathkal<br>Student Paper | 2% |
| 3 | Submitted to Liverpool John Moores University<br>Student Paper | 1% |
| 4 | research.ijcaonline.org<br>Internet Source | 1% |
| 5 | "Computer Vision – ECCV 2016", Springer Nature, 2016<br>Publication | 1% |
| 6 | Submitted to Loughborough University<br>Student Paper | 1% |
| 7 | Submitted to University of Hertfordshire<br>Student Paper | <1% |
| 8 | www.adaface.com<br>Internet Source | <1% |

| | | |
|---|---|---|
| 9 | Submitted to University of North Texas<br>Student Paper | <1 % |
| 10 | www.mdpi.com<br>Internet Source | <1 % |
| 11 | Submitted to Indian School of Business<br>Student Paper | <1 % |
| 12 | Joseph Bamidele Awotunde, Rasheed Gbenga Jimoh, Agbotiname Lucky Imoize, Akeem Tayo Abdulrazaq, Chun-Ta Li, Cheng-Chi Lee. "An Enhanced Deep Learning-Based DeepFake Video Detection and Classification System", Electronics, 2022<br>Publication | <1 % |
| 13 | "Handbook of Digital Face Manipulation and Detection", Springer Science and Business Media LLC, 2022<br>Publication | <1 % |
| 14 | "Computer Vision – ECCV 2020 Workshops", Springer Science and Business Media LLC, 2020<br>Publication | <1 % |
| 15 | "Proceedings of the Future Technologies Conference (FTC) 2020, Volume 1", Springer Science and Business Media LLC, 2021<br>Publication | <1 % |



| | | |
|---|---|---|
| 16 | Submitted to Indraprastha Institute of Information Technology (IIIT-Delhi)<br>Student Paper | <1 % |
| 17 | www.deskera.com<br>Internet Source | <1 % |
| 18 | assets.researchsquare.com<br>Internet Source | <1 % |
| 19 | link.springer.com<br>Internet Source | <1 % |
| 20 | "Communication and Intelligent Systems", Springer Science and Business Media LLC, 2022<br>Publication | <1 % |
| 21 | dspace.ncl.ac.uk<br>Internet Source | <1 % |
| 22 | "Multimedia Forensics", Springer Science and Business Media LLC, 2022<br>Publication | <1 % |
| 23 | Submitted to University of Keele<br>Student Paper | <1 % |
| 24 | Submitted to Sreenidhi International School<br>Student Paper | <1 % |
| 25 | Submitted to Universidade do Vale do Rio dos Sinos<br>Student Paper | <1 % |

| | | |
|---|---|---|
| 26 | fenix.tecnico.ulisboa.pt<br>Internet Source | <1 % |
| 27 | onlineresource.ucsy.edu.mm<br>Internet Source | <1 % |
| 28 | Mahdi Nakhaei, Pouria Nakhaei, Mohammad Gheibi, Benyamin Chahkandi et al. "Enhancing community resilience in arid regions: A smart framework for flash flood risk assessment", Ecological Indicators, 2023<br>Publication | <1 % |
| 29 | arxiv.org<br>Internet Source | <1 % |
| 30 | studenttheses.uu.nl<br>Internet Source | <1 % |
| 31 | Submitted to National University of Ireland, Galway<br>Student Paper | <1 % |
| 32 | Submitted to Queen Mary and Westfield College<br>Student Paper | <1 % |
| 33 | Sai Siddhu Gedela, Nagamani Yanda, Hymavathi Kusumanchi, Suvarna Daki, Keerthika Challa, Pavan Gurrala. "Chapter 57 An Approach toIdentify DeepFakes Using Deep Learning", Springer Science and Business Media LLC, 2023<br>Publication | <1 % |

| | | |
|---|---|---|
| 34 | m.scirp.org<br>Internet Source | <1 % |
| 35 | pdfs.semanticscholar.org<br>Internet Source | <1 % |
| 36 | www.mpi.govt.nz<br>Internet Source | <1 % |
| 37 | Submitted to OP Jindal University, Raigarh<br>Student Paper | <1 % |
| 38 | Mousa Tayseer Jafar, Mohammad Ababneh, Mohammad Al-Zoube, Ammar Elhassan. "Forensics and Analysis of Deepfake Videos", 2020 11th International Conference on Information and Communication Systems (ICICS), 2020<br>Publication | <1 % |
| 39 | Submitted to University of Lincoln<br>Student Paper | <1 % |
| 40 | www.ijnrd.org<br>Internet Source | <1 % |
| 41 | "Biometric Recognition", Springer Science and Business Media LLC, 2019<br>Publication | <1 % |
| 42 | Submitted to University of Greenwich<br>Student Paper | <1 % |

| | | |
|---|---|---|
| 43 | Van-Nhan Tran, Suk-Hwan Lee, Hoanh-Su Le, Ki-Ryong Kwon. "High Performance DeepFake Video Detection on CNN-Based with Attention Target-Specific Regions and Manual Distillation Extraction", Applied Sciences, 2021<br>Publication | <1 % |
| 44 | hdl.handle.net<br>Internet Source | <1 % |
| 45 | idr.nitk.ac.in<br>Internet Source | <1 % |
| 46 | Bhaswati Saha, K. Sai Ram, Jayanta Mukhopadhyay, Aditi Roy, Anchit Navelkar. "Video Based Person Re-Identification by Re-Ranking Attentive Temporal Information in Deep Recurrent Convolutional Networks", 2018 25th IEEE International Conference on Image Processing (ICIP), 2018<br>Publication | <1 % |
| 47 | docplayer.net<br>Internet Source | <1 % |
| 48 | repositorio.unb.br<br>Internet Source | <1 % |
| 49 | www.ijraset.com<br>Internet Source | <1 % |
| 50 | Arun Prakash J., Asswin C.R., Dharshan Kumar K.S., Avinash Dora, Vinayakumar Ravi, | <1 % |

| | | |
|---|---|---|
| | Sowmya V., E.A. Gopalakrishnan, Soman K.P.. "Transfer learning approach for pediatric pneumonia diagnosis using channel attention deep CNN architectures", Engineering Applications of Artificial Intelligence, 2023<br>Publication | |
| 51 | Naimat Ullah Khan, Wanggen Wan, Rabia Riaz, Shuitao Jiang, Xuzhi Wang. "Prediction and Classification of User Activities Using Machine Learning Models from Location-Based Social Network Data", Applied Sciences, 2023<br>Publication | <1 % |
| 52 | Saniat Javid Sohrawardi, Akash Chintha, Bao Thai, Sovantharith Seng, Andrea Hickerson, Raymond Ptucha, Matthew Wright. "Poster", Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security - CCS '19, 2019<br>Publication | <1 % |
| 53 | backend.orbit.dtu.dk<br>Internet Source | <1 % |
| 54 | orca.cf.ac.uk<br>Internet Source | <1 % |
| 55 | unsworks.unsw.edu.au<br>Internet Source | <1 % |
| 56 | www.frontiersin.org<br>Internet Source | <1 % |

| | | |
|---|---|---|
| 57 | www2.mdpi.com<br>Internet Source | <1 % |
| 58 | "Intelligent Learning for Computer Vision", Springer Science and Business Media LLC, 2021<br>Publication | <1 % |
| 59 | "Selfie Biometrics", Springer Science and Business Media LLC, 2019<br>Publication | <1 % |
| 60 | Jalal Ghadermazi, Ankit Shah, Nathaniel Bastian. "Towards Real-time Network Intrusion Detection with Image-based Sequential Packets Representation", Institute of Electrical and Electronics Engineers (IEEE), 2023<br>Publication | <1 % |
| 61 | Jun Yang, Yaoru Sun, Maoyu Mao, Lizhi Bai, Siyu Zhang, Fang Wang. "Model-agnostic Method: Exposing Deepfake using Pixel-wise Spatial and Temporal Fingerprints", IEEE Transactions on Big Data, 2023<br>Publication | <1 % |
| 62 | Sarra Guefrachi, Marwa Ben Jabra, Naif A. Alsharabi, Mohamed Tahar Ben Othman et al. "Deep learning based DeepFake video detection", 2023 International Conference on | <1 % |

| | | |
|---|---|---|
| | Smart Computing and Application (ICSCA), 2023<br>Publication | |
| 63 | Umur Aybars Ciftci, Ilke Demir, Lijun Yin. "FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020<br>Publication | <1 % |
| 64 | cs.montclair.edu<br>Internet Source | <1 % |
| 65 | deepai.org<br>Internet Source | <1 % |
| 66 | idr.l2.nitk.ac.in<br>Internet Source | <1 % |
| 67 | mdpi.com<br>Internet Source | <1 % |
| 68 | www.intechopen.com<br>Internet Source | <1 % |
| 69 | Daniel Stephen, Teddy Mantoro. "Usage of Convolutional Neural Network for Deepfake Video Detection with Face-Swapping Technique", 2022 5th International Conference of Computer and Informatics Engineering (IC2IE), 2022<br>Publication | <1 % |

| | | |
|---|---|---|
| 70 | Nency Bansal, Turki Aljrees, Dhirendra Prasad Yadav, Kamred Udham Singh, Ankit Kumar, Gyanendra Kumar Verma, Teekam Singh. "Real-Time Advanced Computational Intelligence for Deep Fake Video Detection", Applied Sciences, 2023<br>Publication | <1% |
| 71 | Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, Siwei Lyu. "Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics", 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020<br>Publication | <1% |
| 72 | digitalcommons.usf.edu<br>Internet Source | <1% |
| 73 | drum.lib.umd.edu<br>Internet Source | <1% |
| 74 | malariajournal.biomedcentral.com<br>Internet Source | <1% |
| 75 | repository.ju.edu.et<br>Internet Source | <1% |
| 76 | www.cogitatiopress.com<br>Internet Source | <1% |
| 77 | www.ibai-publishing.org<br>Internet Source | <1% |
| | www.scielo.org.mx | |

| | | |
|---|---|---|
| 78 | Internet Source | <1 % |
| 79 | www.springerprofessional.de<br>Internet Source | <1 % |
| 80 | "Information and Communications Security", Springer Science and Business Media LLC, 2020<br>Publication | <1 % |
| 81 | Pavel Korshunov, Sebastien Marcel. "Vulnerability assessment and detection of Deepfake videos", 2019 International Conference on Biometrics (ICB), 2019<br>Publication | <1 % |
| 82 | Dalila Amara, Latifa Rabai. "Comparison of Feature Selection via Semi supervised denoising autoencoder and traditional approaches For Software Fault-prone Classification", Research Square Platform LLC, 2023<br>Publication | <1 % |
| 83 | Kui Zhu, Bin Wu, Bai Wang. "Deepfake Detection with Clustering-based Embedding Regularization", 2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC), 2020<br>Publication | <1 % |

| | | |
|---|---|---|
| 84 | Shraddha Suratkar, Sayali Bhiungade, Jui Pitale, Komal Soni, Tushar Badgujar, Faruk Kazi. "Deep-fake video detection approaches using convolutional – recurrent neural networks", Journal of Control and Decision, 2022<br>Publication | <1% |

| | | | |
|---|---|---|---|
| Exclude quotes | On | Exclude matches | Off |
| Exclude bibliography | On | | |