**Media Engineering and Technology Faculty**
**German University in Cairo**

# Using Facial Emotion Recognition (FER) and Speech Emotion Recognition (SER) to Detect Deepfake Videos

**Bachelor Thesis**

Author: Zeina Hazem Hezzah

Supervisors: Dr. Mervat Mustafa Abuelkheir

Eng. Nada Ibrahim

Submission Date: 19 May, 2024

**Media Engineering and Technology Faculty**
**German University in Cairo**

# Using Facial Emotion Recognition (FER) and Speech Emotion Recognition (SER) to Detect Deepfake Videos

**Bachelor Thesis**

Author: Zeina Hazem Hezzah

Supervisors: Dr. Mervat Mustafa Abuelkheir

Eng. Nada Ibrahim

Submission Date: 19 May, 2024

This is to certify that:

(i) the thesis comprises only my original work toward the Bachelor's Degree

(ii) due acknowledgment has been made in the text to all other material used

---

Zeina Hazem Hezzah
19 May, 2024

# Acknowledgments

First and foremost, I would like to express my deepest gratitude to Dr. Mervat Abuelkheir for her invaluable guidance and support. Her expertise and insightful feedback were instrumental in shaping this research and establishing a mindful approach to tackling the complex problem at hand. I would also like to extend my heartfelt thanks and sincerest appreciation to Nada Ibrahim, without whom this research would not be possible. Her unwavering efforts in helping me overcome various obstacles and adapt my approach countless times to deal with the many limitations of this research area were crucial to the completion of this work. Her innovative ideas and practical advice kept me going, especially during times when I almost gave up. Finally, I would like to thank my family and friends for their constant motivation and encouragement throughout this tedious journey. Their unwavering mental and emotional support, as well as their patience and willingness to tolerate my extensive rants, helped me stay focused and motivated, enabling me to persevere and push through to the finish line.

# Abstract

In the ever-growing age of technology, the circulation of media online has become a cornerstone of daily communication and information dissemination. Amidst this digital era, the proliferation of deepfake technology has raised significant concerns about the authenticity and reliability of digital media. As generation techniques rapidly evolve, synthetic audio and visual content are becoming increasingly indistinguishable from genuine material, rendering traditional deepfake detection methods inadequate. This thesis explores an innovative approach to deepfake detection by exploring the possibility of integrating Facial Emotion Recognition (FER) and Speech Emotion Recognition (SER) to observe the extent to which generation techniques can accurately replicate emotional expressions. The primary aim is to develop a deep learning model capable of distinguishing between real and deepfake videos, evaluating its performance across different emotion classes to identify how discrepancies in emotional expressions can serve as markers for synthetic media detection. Our approach includes utilizing a pre-trained FER model to extract emotion labels and applying pre-processing techniques to highlight essential features in video frames. The developed model combines Convolutional Neural Networks (CNNs) for spatial analysis and Recurrent Neural Networks (RNNs) for temporal analysis, enabling a comprehensive examination of video frames in an attempt to recognize patterns that could be indicative of visual manipulation. The model is separately trained on videos compiled from two deepfake datasets, Celeb-DF and the DeepFake Detection Challenge (DFDC), allowing us to observe its detection capabilities on videos of different qualities and varying contexts. The model's performance is evaluated across different emotion classes to assess its effectiveness in detecting deepfakes and the role emotional expressions play in its detection capabilities. Results show that the proposed model's performance varies by emotion class, with more expressive emotions like fear and disgust being detected more accurately than subtler emotions such as sadness. For the Celeb-DF dataset, the model achieved a maximum testing accuracy of 85.71% on videos expressing fear and a minimum accuracy of 57.73% on videos portraying sadness. Alternatively, testing the model on the DFDC dataset showed a peak accuracy of 83.33% for the disgust emotion class, while videos classified as depicting fear in this dataset achieved a minimal accuracy of 65%. This indicates that emotional cues play a critical role in detecting deepfakes as existing generation techniques struggle to accurately synthesize complex emotional expressions in different contexts. This research underscores the potential of emotion recognition in improving deepfake detection algorithms and highlights the need for continued exploration in this promising field.

# Contents

# Chapter 1

# Introduction

In today's digital era, we are introduced to new technological advances on a near-daily basis. Advancements in artificial intelligence, machine learning, and deep learning have led to the development of new techniques and tools for manipulating multimedia over the years. The proliferation of deepfake technology has presented unprecedented challenges to the authenticity and reliability of multimedia content. Given the rapid advancements in deep generative models and artificial intelligence, synthetic audio and visual media have become so realistic that they are often indiscernible from authentic content for human eyes [18].

Multimedia manipulation is becoming increasingly popular nowadays with almost every video being uploaded to social media platforms leveraging some form of manipulation. Approximately 95 million photos and videos are uploaded daily on Instagram and YouTube, containing some form of media alteration often with the harmless intention of making videos and images more visually appealing for the viewer through the use of filters and editing software [20]. However, a lot of uploaded content utilizes these technologies for malicious purposes such as defamation, political sabotage, blackmail, disseminating fake news, and terrorist propaganda [2].

Deepfakes are videos that have been edited to alter the identity of the depicted person through facial or speech manipulation. They are often generated by deep learning techniques and are becoming increasingly difficult to detect [13]. This phenomenon poses significant threats to various domains, including politics, journalism, entertainment, and beyond, by undermining trust, manipulating public opinion, and potentially causing irreparable harm to individuals and institutions.

In the age of social media, platforms like TikTok and Instagram have revolutionized user-generated content through their innovative assortment of filters and effects, enabling users to creatively edit, modify, and personalize their videos. Multiple AI-based trends have gone viral on TikTok, with a variety of filters allowing users to turn themselves into cartoon characters [16], edit celebrity faces onto their own [28], and generate images or art pieces by simply entering a prompt through the filter [1].

However, the increased availability of these technologies has also inadvertently fueled the rise of deepfake and AI generation, making it possible for almost anyone to create fake content with just the click of a button. With TikTok filters and similar features becoming increasingly intuitive and user-friendly, social media users are now faced with the unsettling reality that deepfake content is not only more accessible but also more prominent than ever, raising serious concerns regarding authenticity, misinformation, and privacy in the digital age.

## 1.1 Motivation and Objectives

Preventing and reducing the spread of deepfakes has become a crucial task in protecting the authenticity of digital material and maintaining confidence in media sources. Traditional methods of detecting deepfakes mainly depend on forensic analysis, which involves identifying irregularities in pixel patterns or defects caused during the editing process. However, with the advancement of deepfake technology, these techniques are becoming less reliable, and more innovative approaches are needed to differentiate between synthetic content and genuine media.

One promising approach for addressing this challenge lies in the domain of emotion recognition. Emotions play a crucial role in human communication, influencing facial expressions, vocal intonations, and body language. However, deepfakes often fail to accurately replicate the nuanced emotional cues exhibited by authentic individuals, resulting in discrepancies that can be leveraged for detection purposes. Researchers have been exploring the potential of machine learning and computer vision techniques to identify the subtle indicators of deception present in deepfake content through emotion recognition algorithms.

Motivated by the potential that lies in this field, this thesis aims to:

- Develop a deep learning model capable of distinguishing between authentic and deepfake videos.
- Explore and evaluate different datasets for training and testing the deepfake detection model.
- Implement different pre-processing techniques to extract relevant features from video data and investigate how they affect the model's performance.
- Conduct comprehensive experiments to assess the robustness and generalization capabilities of the developed model across different datasets.
- Exploit previously developed emotion recognition models in an attempt to identify emotions exhibited by the subjects in real and deepfake videos.
- Assess the ability of deepfake generation techniques to accurately replicate human emotions.

- Observe the model's ability to detect deepfake videos exhibiting different major emotions
- Evaluate the designed model and address potential vulnerabilities and limitations of the model.

## 1.2 Outline

The thesis is structured as follows:

- In Chapter 2, we discuss the background information relevant to this study. It provides an overview of the threats and impact of Artificial Intelligence (AI)-generated media. It also highlights previous efforts that have contributed to this research area, observing previously developed emotion recognition models and different approaches to detecting deepfake videos.
- In Chapter 3, the methodology of this study is outlined, detailing all the procedures taken to address our objectives. First, the chapter clearly establishes the problem statement and the general approach followed throughout the study. Then, we provide a brief description of the datasets used, namely the Celeb-DF and DFDC datasets, and the pre-processing steps implemented to prepare the data for model training. We also outline the emotion recognition process used to extract emotion labels from the videos in each dataset. We then discuss the architecture of the developed model, examining each component, and the preparation of data for training and testing the model.
- In Chapter 4, the results obtained from emotion extraction as well as model training are presented, providing a detailed analysis of the metrics and accuracies used to evaluate the model's efficiency. We first discuss the distributions of emotions across the videos in each dataset, examining how detected emotions differ between real and deepfake videos. Next, we describe the training process for each dataset used, examining the model's training and validation accuracies for both. We then analyze the way the model performs on different emotion classes and discuss the findings, comparing the model's accuracies in predicting deepfake videos exhibiting different perceived emotions. The limitations encountered throughout the study are also highlighted in the chapter.
- Finally, Chapter 5 concludes the thesis, summarizing the overall results of the study and exploring their significance. It also suggests possible areas of future research that require further investigation, highlighting the potential for advancements in this field.

# Chapter 2

# Background

With the rise of new deepfake generation technologies and the growing threats they pose, interest in the research area has recently grown. This chapter discusses previous work and the available literature in the field of deepfake detection. First, in Section 2.1, the concept of deepfake videos and their origins are explained. In Section 2.2, we discuss the ethical implications and societal responses to manipulated media by highlighting recent scandals involving AI-generated content and their impacts. Section 2.3 discusses relevant studies done in the field of emotion recognition and the approaches used to detect expressions in media. Finally, Section 2.4 addresses previous attempts at developing deepfake detection models with some researchers following a uni-modal detection approach and others opting for multi-modal approaches.

## 2.1 An Overview of Deepfakes

A combination of "deep learning" and "fake", deepfakes are hyper-realistic synthetic media, generated using advanced AI algorithms and digitally manipulated to depict people saying and doing things that never actually happened by seamlessly superimposing one person's features onto another's [38]. A deepfake is simply "material created by Deep Learning (DL) that seems genuine in a human's eyes", as defined by Heidari et al. [4].

This term was coined in late 2017, after an anonymous Reddit user, under the pseudonym "deepfake", used deep learning methods to create and upload photo-realistic fake pornographic videos, created by replacing a person's face with another person's face, claiming that they belonged to famous actresses such as Taylor Swift, Scarlett Johansson, Aubrey Plaza, Gal Gadot, and Maisie Williams [2].

## 2.2 Recent Impact of Deepfake Technologies

Since 2017, AI media generation has grown more popular, and manipulated media content has circulated the internet at an alarming rate. Recently, major scandals and controver-

sies involving deepfake manipulation have caused recent uproar regarding the ethical implications and threats they impose.

In February 2024, an elaborate financial scam involving a deepfake conference call cost a major multinational firm $25.6 million after an employee at the Hong Kong office was duped into transferring the money to an offshore account, believing he was receiving instructions from the company's chief financial officer. The worker revealed that although the request seemed suspicious, he set his doubts aside after the conference call was attended by what he thought were several recognizable staff members and co-workers. He explained that the people on the call "looked and sounded just like colleagues he recognized" and he only realized that all the faces he saw were fake after personally inquiring about the transfer at work [11].

In January 2024, another major scandal involving American pop musician, Taylor Swift, generated significant public outrage and prompted serious discussions about the dangers of AI-generated content. The scandal saw a series of sexually explicit synthesized images and videos of the singer massively circulating on X (formerly Twitter), with one post lasting on the platform for approximately 17 hours, receiving more than 45 million views, 24,000 reposts, and thousands of likes before being taken down [30]. Despite the rallying efforts of countless fans and supporters of Taylor Swift (referred to as "Swifties") to get the pornographic content removed, the pervasive nature of the internet meant that the content spread rapidly and was nearly impossible to get rid of. Alternatively, Swifties attempted to bury the pornographic content with an outpouring of "Protect Taylor Swift" comments and hashtags until searching the term "Taylor Swift" was completely banned on the platform after realizing the seriousness of the situation. This incident highlighted the potential for deepfakes to cause severe emotional, financial, and reputational harm, particularly to women and young girls, who are frequently targeted by such malicious content. The scandal not only sparked public debate but also prompted lawmakers to take legislative action to combat the threats posed by deepfakes and AI technologies. U.S. Representative Joe Morelle and other politicians called for new legislation to criminalize the creation and distribution of non-consensual deepfake pornography, which he explained "can cause irrevocable emotional, financial, and reputational harm", particularly to women and young girls, who are frequently targeted by such malicious content [33].

## 2.3 Emotion Recognition

Several studies have observed different means of emotion prediction and recognition. The field of emotion perception is centered around two seemingly contradictory hypotheses: The categorical theory, which asserts the existence of six fundamental, unique emotions (happiness, anger, sadness, surprise, disgust, and fear), and the dimensional theory, which suggests the existence of two dimensions of emotional space, valence and arousal, within which a spectrum of emotions exist [25].

### 2.3.1 Categorical Emotion Recognition

One approach used to predict emotions is the categorical approach, in which emotions are classified into distinct classes. Given an input video, the features are analyzed and labeled with the specific emotion class to which they most closely belong.

Mittal et al. [29] followed this approach when creating their model, where the perceived emotion embeddings extracted from the features were classified into one of 6 discrete emotions defined by the Ekman Model: happy, sad, angry, fearful, surprise, disgust. Emotions that didn't fit into any distinct class were labeled as "neutral". They used the CMU-MOSAI dataset [5] to train their model, as it contains a multitude of videos and audio expressing emotions that fall into these classes.

Similarly, the model proposed by Conti et al. [13] adopted a categorical prediction approach, as the speech feature vectors extracted from the SER model were input to a Random Forest binary classifier to estimate the class y to which the input signal x belongs. They trained and tested their model on the IEMOCAP dataset [10], classifying emotions into one of four discrete classes: angry, happy, sad, and neutral.

### 2.3.2 Dimensional Emotion Recognition

Alternatively, some models follow a dimensional approach for predicting emotions, viewing emotions as combinations of variable metrics rather than discrete categories.

For instance, the model proposed by Hosler et al. [17] was built around estimating values for quantitative features of emotions and defining the perceived emotions in terms of those values rather than labeling them with distinct terms. Their model primarily focused on the valence and arousal features of emotions, with each predicted emotion falling somewhere on the 2D spectrum formed by the two dimensions. This allowed them to observe more nuanced emotions; hence, recognizing subtle differences and training their model accordingly.

## 2.4 Deepfake Detection

Several different approaches have been proposed to detect deepfake videos, varying in different aspects such as the observed modalities and the detection model architectures.

### 2.4.1 Uni-Modal Deepfake Detection

Various models have been developed within the past few years in an attempt to detect deepfakes focusing on audio and video modalities independently, processing visual and auditory channels to recognize anomalies in facial appearance and speech tonalities, respectively.

Conti et al. [13] focused on the audio component of deepfakes by proposing a method to identify whether a given speech signal is real or deepfake using sentiment analysis through leveraging semantically meaningful audio embeddings. Their approach focused on synthetic audio generated through Text-to-Speech (TTS) synthesis and Voice Conversion (VC) techniques. They developed a transfer-learning model consisting of a Speech Emotion Recognition (SER) system and a Synthetic Speech Detector (SSD). Semantic. Semantic speech features were extracted from the speech signal, by estimating the expressed emotion using the SER, and then fed through the SSD which used a Random Forest Classifier to associate the input features with some emotion class $y$ to which it belongs. The emotion features were extracted and pre-processed by computing the Short Time Fourier Transform (STFT) in the Mel-Frequency Domain and applying a logarithmic transform to obtain a log-mel spectrogram. The first and second derivatives were then computed along the frequency domain. The matrix formed by stacking the spectrogram and its derivatives was standardized by employing z-score normalization to obtain the final 3D matrix. The pre-processed input was then passed through a set of 3D convolutional layers, a linear layer, a BLSTM and attention layer, and a sequence of dense layers. The network outputs a probability value for each emotion class, signifying the extent to which the emotion belongs to each class, which serves as the basis for classification. The model was trained on a variety of datasets including ASVspoof 2019 [39], LibriSpeech 2019 [32], Cloud2019 2019 [23], and Interactive Emotional Dyadic Motion Capture 2019, IEMOCAP [10], each augmented with white noise to assess the model's robustness against audio deterioration. Their system outperformed existing CNN-based methods, reaching an average Area Under Curve (AUC) of 0.98 for almost all tested datasets. The addition of white noise to the training set did not significantly affect the detector's performance on a noisy testing set. However, the system trained on clean data achieved higher accuracy on clean testing data, suggesting that for high Signal-to-Noise Ratio (SNR) levels, training on clean data is more advantageous than using augmented data.

Kumar et al. [20] explored the detection of deepfakes on a visual basis by observing alterations in images and videos using a learning-based algorithm. They proposed a deep learning-based network for detecting reenacted frames in videos, based on localized features extracted by the Face2Face reenactment approach [35], and a loss function for balanced training of streams in the proposed network. Face detection was performed using the S3FD approach [40], and the region of interest was extracted using strict square cropping centered around the face to eliminate as much of the background as possible. They proposed a multi-stream network consisting of five parallel ResNet-18 models, each with two outputs, concatenated to form a 10-dimensional vector, as shown in Figure 2.1. This vector was then passed to a fully connected layer for binary classification. An input image X is cropped into five smaller images and fed through the network, which outputs a binary value Y with 0 signifying original and 1 signifying altered frames. The loss function was minimized throughout the training process to avoid the model developing a bias for a particular ResNet model. They trained and tested their model using the FaceForensics Source-to-Target reenactment dataset [34] consisting of 1004 unique videos from YouTube, modified using the Face2Face approach. Different sections

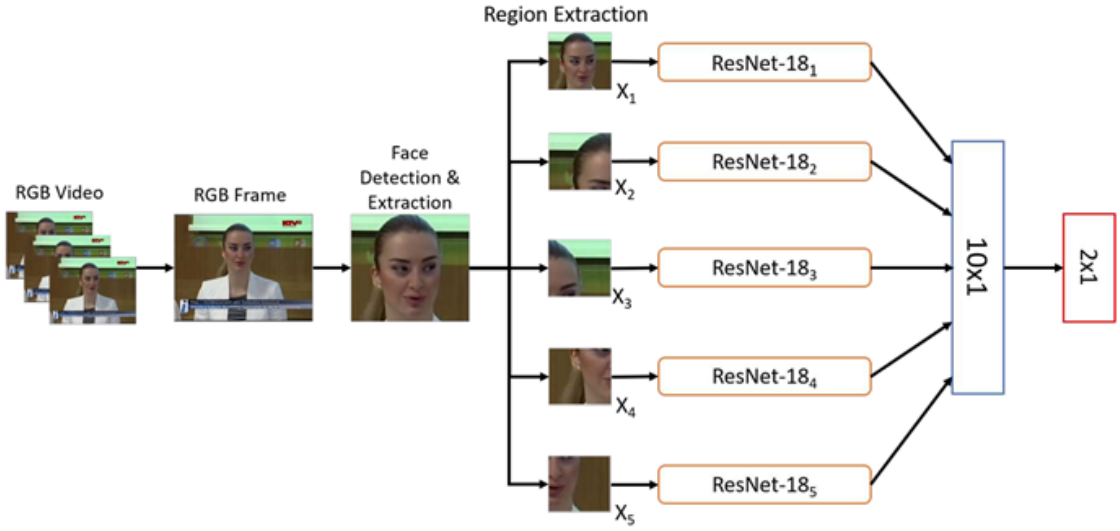



Figure 2.1: Proposed pipeline by Kumar et al. [20]

of the face provided different classification accuracies; for instance, left-sided features were found to perform better than right-sided features, and the eye region gave more consistent classification performance than the cheek region. Their algorithm yielded a mean classification accuracy of 90.40%.

### 2.4.2 Multi-Modal Deepfake Detection

To the best of our knowledge, limited studies have explored the effect of utilizing multiple modalities in tandem to detect deepfakes, rather than just approaching the problem from a single modality.

Mittal et al. [29] developed a method for detecting deepfake videos that analyzes both facial and speech modalities, comparing the perceived emotion for each modality. The method utilized a Siamese network-based architecture, in which the network was composed of multiple sub-networks that were all trained with the same parameter values. Additionally, it used a triplet loss function to model the similarities between the audio-visual modalities and emotions, to distinguish whether the input video is real or deepfake. Their model exploited the correlation between visual and audio modalities extracted from the same video by examining the affective cues in both modalities to classify "real" and "fake" videos. Their approach relied on a combination of CNNs to extract modality embeddings, encoding the semantic meaning of the digitized features as mathematical vectors, and Long Short-Term Memory (LSTM) layers to detect the perceived emotion embeddings for the features. To train the model, similarity scores were computed by fixing the modality that was manipulated less and calculating the Euclidean distances between it and each of the real and fake embeddings for the other modality. The same was

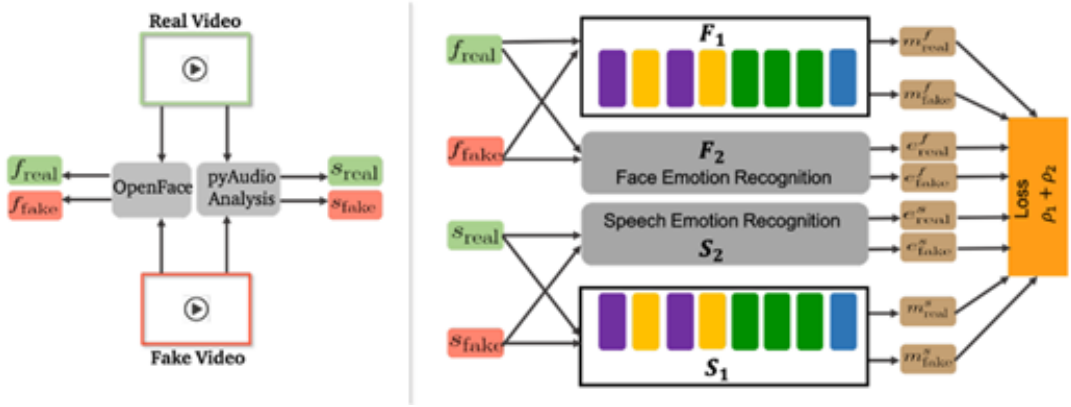



Figure 2.2: Mittal et al. [29] Multi-modal Categorical Detection Model

done for the perceived emotion embeddings and then both scores were added up to derive the cumulative loss that was propagated back through the networks to update parameter values, as shown in Figure 2.2. Through training, an average threshold value was obtained which was later used as a comparison metric while testing to classify videos as real or fake. Their approach achieved an AUC score of 84.4% on the DeepFake Detection Challenge (DFDC)) dataset [6], which is around 9% higher than most state-of-the-art methods, and 96.6% on the DeepFake-TIMIT [19] dataset.

Hosler et al. [17] proposed an alternate multi-modal deepfake detection method based on valence-arousal emotion analysis. They followed a dimensional approach to detecting emotions by observing the values of quantitative features, such as valence—the emotional affect (positive or negative), and arousal—the emotional intensity (excited or calm), rather than classifying emotions into distinct classes. This continuous model allowed them to evaluate more subtle, inconsistent emotions that cannot be uniquely categorized. Given an input video, face and speech Low-Level Descriptors (LLDs) were first extracted using the facial action unit extractor of OpenFace [3] and the speech-based feature extractor of OpenSmile [15], respectively. A sequence of LLDs for each modality was obtained by running the programs on multiple time frames of the video. The face and speech LLD time series were then input through an LSTM-based neural network, trained using the SEMAINE database [26], that returns two time series: valence evolution and arousal evolution over time for the input modality. Once all 4 valence and arousal signals ($\text{VAL}_{\text{speech}}$, $\text{AR}_{\text{speech}}$, $\text{VAL}_{\text{face}}$, $\text{AR}_{\text{face}}$) were obtained, deepfake detection was performed by analyzing a set of statistical metrics to observe the relationships between signals. For instance, Lin's Concordance Correlation Coefficient (CCC) was used to measure the correlation between both valence signals and that between both arousal signals to see how closely the facial emotions matched those conveyed through speech. The measured statistical features were then concatenated and passed to a supervised classifier to distinguish if the video was real or deepfake. Moreover, in an attempt to exploit the temporal evolution of valence and arousal values, a learning-based detection approach was also proposed to classify videos by passing the valence and arousal time series to a

lightweight LSTM, rather than a static supervised classifier. All observed classifiers (supervised and LSTM-based) were trained on the DeepFake Detection Challenge (DFDC) training dataset. The study found that speech-derived features were detected with a much higher accuracy (around 85%) than facial features (51.8%) which suggests that naturally expressive speech, with the same emotional range as authentic speech, is significantly more difficult to synthesize than facial emotion features, since most deepfake algorithms preserve the original speaker's instantaneous expressions, making them more convincing. However, following the learning-based detection approach utilizing LSTMs, the accuracy rose to around (99.5%) which indicates that although synthetic faces may be able to recreate instantaneous emotions, they are not temporally consistent.

# Chapter 3

# Methodology

The following chapter discusses the implementation process of the overall project. Firstly, we explain our general approach to developing and evaluating a deep learning model to detect deepfake videos in Section 3.1. Then, we discuss the datasets used to train and evaluate the model in Section 3.2. Section 3.3 explains how emotion recognition is leveraged to extract major emotion labels for the videos in the datasets. Next, in Section 3.4 we break down how the data was pre-processed and augmented before being input to the model. Finally, Section 3.5 explains the full architecture and configuration of the developed deep learning model.

## 3.1 General Approach

The main objective of our study is to develop a deep learning model capable of classifying a given input video containing a person's face as either real or deepfake by analyzing spatial and temporal features in the video, furthermore exploring the effect different perceived emotions have on the model's performance capabilities .

Tackling the problem of deepfake detection requires a careful selection of datasets necessary for training and evaluating the model's accuracy. Given the novelty and complexity of the research area, there are limited datasets available capturing a diverse range of authentic and manipulated videos suitable for training a detection model. After browsing through a selection of accessible datasets, two main datasets, Celeb-DF [21] and the Deepfake Detection Challenge (DFDC) dataset [6], which are further discussed in Section 3.2, were selected for training our model.

A general overview of the approach adopted throughout this study involves using a pre-trained FER model to extract emotion labels indicating the major emotions expressed in both deepfake and real videos as outlined in Section 3.3, then meticulously pre-processing the synthesized deepfake videos and their authentic counterparts to extract relevant features as explained in Section 3.4. A hybrid model architecture is developed as outlined in Section 3.5, combining the capabilities of Convolutional Neural Networks (CNNs) and

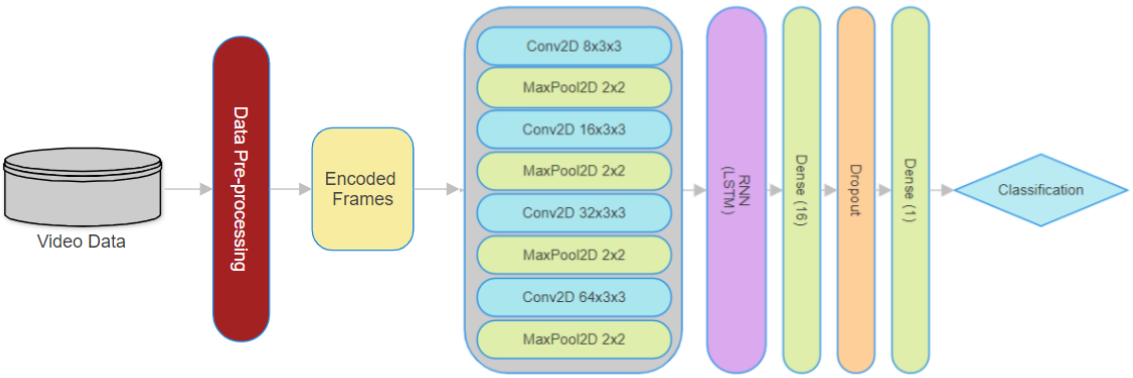



Figure 3.1: Flow of data throughout the model

Table 3.1: Deepfake datasets considered for the study. The selected datasets are highlighted in red

| Dataset | Number of Videos | | | Modalities | |
|---|---|---|---|---|---|
| | Real | Fake | Total | Video | Audio |
| DeepfakeTIMIT [19] | 0 | 640 | 640 | Yes | Yes |
| Celeb-DF [21] | 590 | 5,639 | 6,229 | Yes | No |
| FaceForensics++ [34] | 1,000 | 4,000 | 5,000 | Yes | No |
| DFDC [14] | 23,654 | 104,500 | 128,154 | Yes | Yes |

Recurrent Neural Networks (RNNs) to effectively analyze spatial and temporal relations between the extracted features and learning how they contribute to distinguishing between real and manipulated deepfake videos. Figure 3.1 shows the general flow of data throughout the model development process providing a brief overview of the model and the proposed approach.

## 3.2 Datasets

This section explores the different datasets used to train and evaluate our model. The choice of a suitable dataset for the problem at hand is crucial to the efficacy of our model. In our case, we required a comprehensive dataset containing a collection of manipulated deepfake videos along with their authentic unedited counterparts. Since this is a relatively new research area, there weren't many readily available datasets to choose from. We explored various datasets, whose attributes are outlined in Table 3.1, settling on two main datasets to train our model: Celeb-DF [21] and the Deepfake Detection Challenge (DFDC) dataset [6] which are further discussed in Sections 3.2.1 and 3.2.2, respectively.


Figure 3.2: Examples of Real and Deepfake videos from the Celeb-DF dataset [27]

### 3.2.1 Celeb-DF

The Celeb-DF dataset is a comprehensive collection of videos assembled specifically for research on detecting deepfake videos. The dataset was particularly designed to focus on detecting deepfake videos involving the manipulation of celebrities' faces; therefore, it contains a multitude of closeup videos of various celebrities from interviews and talk show segments.

The dataset is comprised of 590 original videos sourced from YouTube, featuring 59 subjects of different ages, ethnic groups, and genders, along with 5639 corresponding deepfake videos generated using a variety of facial manipulation techniques. The distribution of real and deepfake videos across the dataset can be seen in Figure 3.4a. The videos are classified into two folders: "Celeb-real" and "Celeb-synthesis", containing the authentic videos and the synthesized deepfakes, respectively. For each subject, there are approximately 9-12 real videos filmed under various conditions, hence providing a diverse and representative collection of data. The average length of all videos is around 13 seconds, with a standard frame rate of 30 frames per second. The deepfake videos are generated by swapping faces for each pair of the 59 subjects using an improved DeepFake synthesis algorithm [22]. Some examples of the videos in the Celeb-DF dataset are shown in Figure 3.2.

### 3.2.2 DeepFake Detection Challenge (DFDC)

The DeepFake Detection Challenge (DFDC) was an initiative launched by Facebook in collaboration with various partners, including AWS and Microsoft, aimed at encouraging research and development in the detection of deepfake videos. The competition provided participants with one of the largest collections of real and face-swapped deepfake videos,

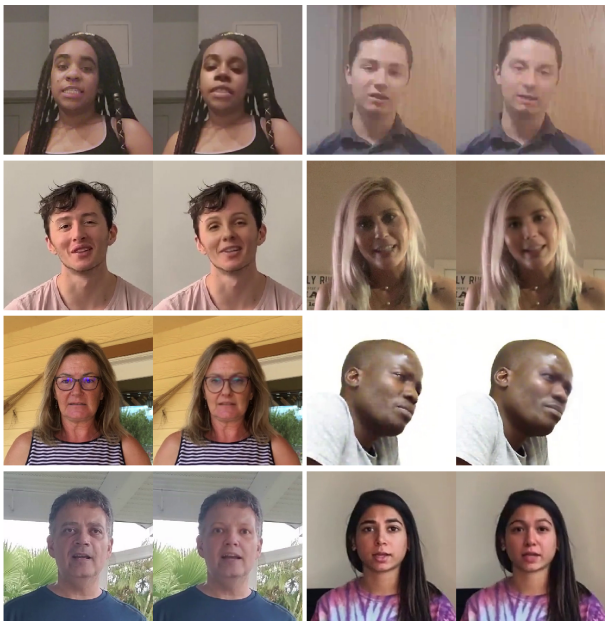


Figure 3.3: Examples of Real and Deepfake videos from the DFDC dataset [8]

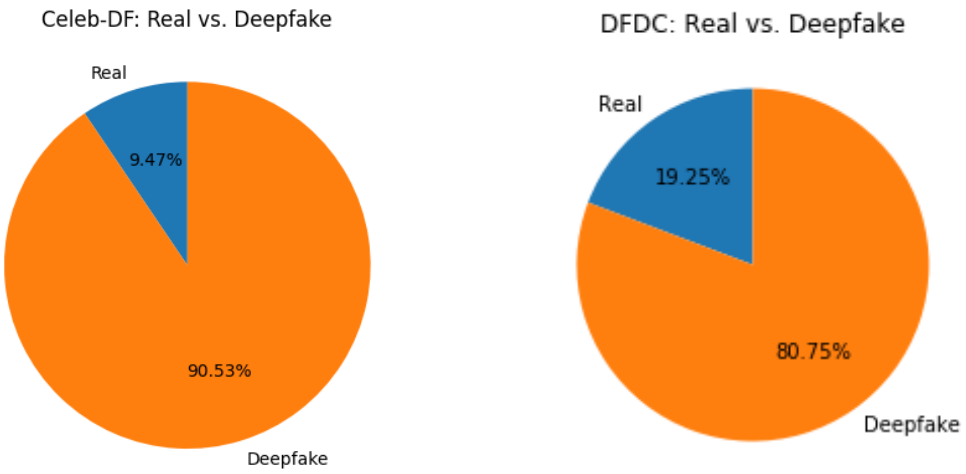
with over 120,000 clips sourced from 3,426 paid actors, generated using several deepfake and GAN-based methods. A few examples of the videos included in the DFDC dataset can be seen in Figure 3.3.

Due to the exceptionally large size of the dataset and limited processing capabilities, a smaller preview sample was provided consisting of 4,113 deepfake videos generated using two facial modification algorithms applied to 1,131 original videos of 66 consenting individuals of various genders, ethnic groups, and ages [8].

The accessible sample of the dataset was divided into two folders: "train_sample_videos" and "test_videos", each containing 400 videos designated for training and testing the model, respectively. Additionally, a metadata JSON file accompanied the training videos which indicated the label for each video ('REAL' or 'FAKE'). However, the testing data was, unfortunately, unlabeled since the dataset was part of a competition and the organizers likely wanted to ensure that participants would not directly train their models on the test data to manipulate the model's testing accuracy. Therefore, we relied solely on the videos in the training sample, further splitting those into training, testing, and validation sub-samples as discussed in Section 3.4.1. The distribution of real and deepfake videos across the used sample of the dataset can be seen in Figure 3.4b.

## 3.3 Emotion Recognition

In our endeavor to analyze the extent to which deepfake videos can be perceived as real videos, we dive into the field of emotion recognition to observe how well deepfakes can mimic natural expressions. The aim is to study the detection of emotional expressions




(a) Celeb-DF (b) DFDC

Figure 3.4: Distribution of Real and Deepfake videos in Celeb-DF (a) and DFDC (b)

in deepfake videos and how they vary from authentic expressions to examine the extent to which deepfake generation techniques can simulate emotions. Two ares of interest when it comes to emotion recognition in videos are Speech Emotion Recognition (SER) and Facial Emotion Recognition (FER). This section discusses how both modalities were explored throughout the considered datasets in an attempt to extract useful information for deepfake detection.

### 3.3.1 Speech Emotion Recognition (SER)

Initially, our appoach to detecting deepfake videos involved exploring both audio-visual modalities, using SER models to analyze emotional cues in the audio. However, this approach faced significant limitations due to the availability and quality of suitable datasets. As shown in Table 3.1, the number of available datasets used for deepfake research is very limited due to the novelty of the research area. Moreover, very few datasets contained both video and audio modalities, hindering our ability to follow through with the planned multi-modal approach.

The DFDC dataset [6], although containing audio, presented challenges as the audio quality was often unclear and not expressive enough for SER models to effectively detect emotional nuances. This lack of clarity and expressiveness in the audio meant that incorporating SER into the model would not provide significant performance improvements.

The DeepfakeTIMIT dataset [19] included audio data but only contained deepfake videos, with no real videos for comparison. This absence of real videos prevented the creation of a balanced training set necessary for the model to learn to distinguish between real and deepfake content. Without a basis for comparison, the dataset could not be used to effectively train our model to detect deepfakes.

These limitations in dataset availability and quality led to the decision to focus solely on the visual modality for deepfake detection, as the existing datasets did not support a robust and effective implementation of SER for this purpose.

### 3.3.2 Facial Emotion Recognition (FER)

Due to the limitations imposed by the datasets and the inefficiency of exploring SER, we chose to focus our study on the use of Facial Emotion Recognition (FER) to explore how the model performs on different expressed emotions. Despite many advancements in deepfake generation techniques, many synthesized videos fail to recreate facial expressions as accurately as they are naturally portrayed; therefore, analyzing the detection model's performance on each emotion class is a promising approach that provides significant insight into which emotions are most easily detectable and which are easily replicated.

#### FER Model

To extract emotion labels from the videos in both datasets, we used the pre-trained FER model developed by Madhur Chhajed [12]. The model follows a CNN architecture and is trained on the Extended Cohn-Kanade (CK+) dataset [24], which contains seven classes of emotions (anger, contempt, disgust, fear, happiness, sadness, surprise), achieving a testing accuracy of 98%. The model is designed to take an input of either a live video stream or a video file and detect the emotion expressed in each individual frame as outlined in the system flow chart shown in Figure 3.5.

#### Emotion Extraction

To extract the dominant emotion expressed in each video, the Chhajed FER model [12] is first imported and loaded using the Tensorflow Keras library. We chose to focus our observations on only 6 of the 7 emotions defined by the model, neglecting 'contempt' due to its possible ambiguity in the context of deepfake detection. Therefore, a new emotion dictionary was defined to map the model's categorical integer output to its corresponding emotion label as follows:

Table 3.2: Emotion Dictionary used for FER Model Predictions

| Model Output | Predicted Emotion Label |
|---|---|
| 0 | anger |
| 2 | disgust |
| 3 | fear |
| 4 | happiness |
| 5 | sadness |
| 6 | surprise |

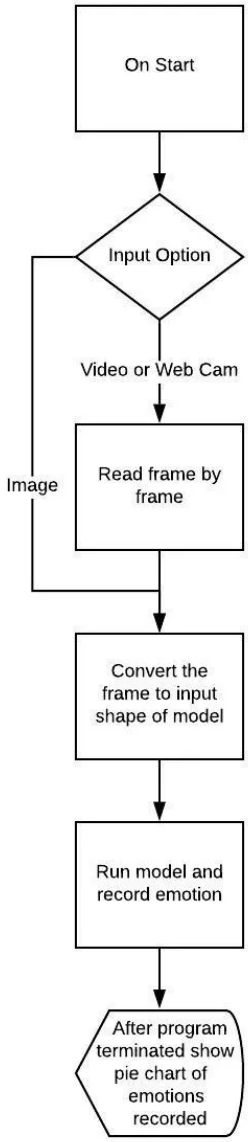



Figure 3.5: Flow of Chhajed FER System [12]

For each of the two datasets, we start by creating a data frame to store the extracted information for each video, namely the file name, the major emotion portrayed throughout the video, and the video label ('REAL' or 'FAKE'). This ensures easy access to all necessary data about each video in future processing steps.

For each video, the OpenCV library [7] is used to capture individual frames. Due to the large number of frames captured from each video and limited processing capabilities, every 10th frame is processed while the rest are skipped. Since variations between consecutive frames are very subtle, selecting every 10th frame provides a representative sample of the video, while avoiding the redundancy of similar frames, hence reducing the number of frames required to process.

Each frame is then pre-processed before being fed to the model to ensure it fits the input shape and requirements of the model. The frames are pre-processed following the same steps executed while training the model to ensure the most accurate predictions. The pre-processing steps include:

1. Using the Haar Cascade Classifier [37] to detect faces in a grayscale version of the frame and cropping it around them
2. Resizing the cropped frame to $48 \times 48$ to be accepted by the model layers
3. Encoding the frame features as an array of pixel values
4. Normalizing the features by dividing all values by 255

Once the frame is properly processed, it is input to the FER model, which returns an array of percentages representing the probability of the frame exhibiting each of the 7 emotions. The index of the highest percentage value is mapped to the defined emotion dictionary to determine the major emotion class of the given frame. For each video, a dictionary of emotion counts is defined to keep track of the number of frames exhibiting each of the 6 emotions. Once the emotion of a frame is predicted, the count of that emotion in the dictionary is incremented accordingly.

After extracting emotions from all selected frames of a given video, the major emotion portrayed throughout the video is determined by selecting the emotion label with the highest number of frames in the previously defined emotion count dictionary. Next, the label of each video ('REAL' or 'FAKE') is generated as detailed in Section 3.5.2 depending on the dataset being processed.

The video name, the extracted major emotion, and the generated label are then appended to the data frame as a new entry and the process is repeated for every video until the dataset is fully processed and the emotions of all videos are extracted. Finally, the data frame corresponding to each dataset is saved to an external CSV file to avoid having to repeat the process every time the dataset is used.

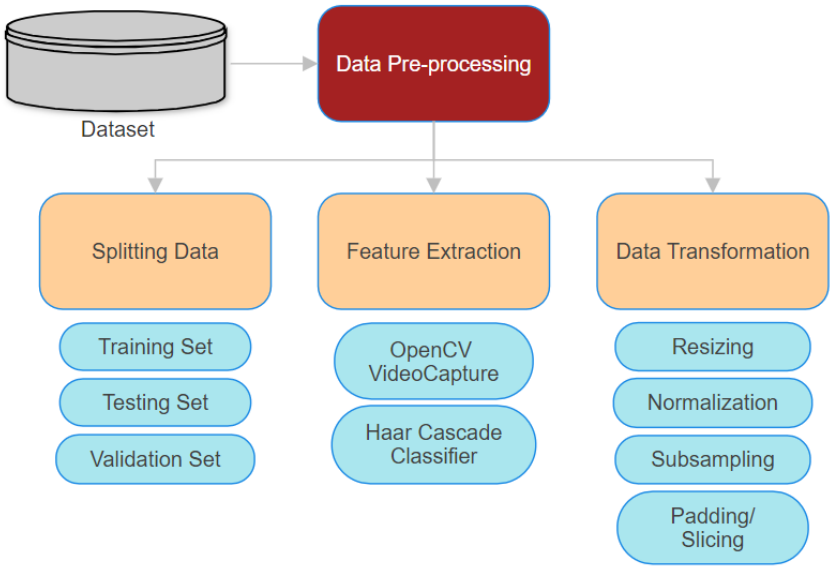



Figure 3.6: Data Pre-processing flow

## 3.4 Data Pre-processing

Before using either dataset, we performed some essential data pre-processing to extract relevant features and adjust all the given data to fit the input specifications of our model. This section explains the steps taken to modify the videos retrieved from the datasets before feeding them to our model as outlined in Figure 3.6.

### 3.4.1 Splitting Data

To ensure the robustness of our deep learning model and its ability to generalize its performance given unknown data, the datasets are divided into subsets for training, testing, and validation. The test_train_split() function from the scikit-learn library [9] is used to generate random subsets from the initial datasets.

- Celeb-DF: Given the large size of the dataset (5639 fake and 590 real videos), using all the given data requires long processing times and computational costs. Therefore, the dataset is initially reduced by splitting each category of files (real and fake) in half, producing 2819 fake and 295 real videos. Next, the testing set is generated by further splitting each category of the reduced dataset with a factor of 0.2 (20%), yielding 564 fake and 59 real videos. Once again splitting the remaining data by a factor of 0.1 (10%), 226 fake and 24 real videos make up the validation set. The remainder of the reduced dataset (2029 fake and 212 real) is designated as the training set. Finally, the subsets of real and fake files are concatenated to form the training, testing, and validation sets containing both real and fake files.

To prevent any inherent biases that may arise from the original ordering of the files in the dataset, the training set is randomly shuffled.

- DFDC: The sample of the dataset used contains 400 labeled videos which are divided to form the training, testing, and validation sets. Upon examining the metadata.json file included with the videos, the dataset is found to contain 323 fake and 77 real videos. To ensure that each subset of files contains an even distribution of fake and real videos, the metadata file is first analyzed to divide the list of videos into separate real and fake lists since they are all initially grouped in one directory. Each list is then split with a factor of 0.2 (20%) to create the testing set lists (65 fake and 16 real). The remaining data is then split once again with a factor of 0.1 (10%) to make up the validation set lists (26 fake and 7 real), while the rest of the data forms the training set lists (232 fake and 54 real). The real and fake lists for each subset are then concatenated to form the three lists: *train_files, test_files, and val_files*, comprising a total of 286, 81, and 33 videos, respectively. Finally, the training set is shuffled to reduce bias and eliminate any patterns that can be memorized by the model during training.

### 3.4.2 Feature Extraction

The video files are processed to collect necessary information by extracting frames from each video and cropping them around faces to focus on relevant data and eliminate any background noise.

- Capturing Frames: The frames of each video are read using the OpenCV environment's 'VideoCapture' class [7]. Each frame is then further processed before being appended to an array collecting the frames extracted from the given video

- Face Detection and Cropping: The Haar Feature-based Cascade Classifier of the OpenCV library [37] is used to detect faces in each frame and the detected regions are then cropped out to eliminate as much of the background as possible, providing more accurate data.

### 3.4.3 Data Transformation

A series of transformation steps are applied to the extracted frames to ensure all the data is uniformly shaped and encoded allowing the model to process them more effectively.

- Resizing Frames: Each frame is resized to a specified image size of $64 \times 64$ to ensure that all frames have the same size creating a consistent input stream for the model.

- Normalization: The pixel values of each frame are normalized to a range between [0, 1] by dividing by 255.0. This step helps ensure numerical stability during training by reducing large variations in the input values to the model, therefore allowing it to train more effectively.

- Subsampling: To reduce the computational cost of processing video data, every 10th frame from the extracted frames for each video is selected. This step is prompted by the large number of frames generated during the extraction process, which can result in lengthy inputs that are computationally expensive. Selecting a subset of frames helps reduce processing time and computational costs while still maintaining a representative sample of the data and retaining temporal information.

- Padding/Slicing: Selecting every 10th frame from approximately 10-15 second videos with a frame rate of 30 fps results in around 30-45 frame samples per video. To ensure consistent input lengths, each set of frames is adjusted to a chosen sequence length of 50 frames. In the cases where the number of processed frames is fewer than 50, sequence padding is applied by replicating the last frame until the desired sequence length is reached. Alternatively, if a video produces more than 50 frames, the sequence is sliced by selecting a random subset of consecutive frames equal to the desired sequence length. This processing step ensures that all frame sequences are the same length therefore increasing the consistency and uniformity of the input streams.

## 3.5 Model

This section outlines the deep learning model that was developed throughout this thesis and trained on the Celeb-DF and DFDC datasets. Firstly, the model architecture and configuration are broken down in Section 3.5.1. Then, we explain how the data is labeled and compiled into suitable input streams for the model in Section 3.5.2.

### 3.5.1 Architecture

Since the topic at hand requires analyzing sequential and visual data inherent in video files, our proposed model combines the capabilities of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to effectively identify the shortcomings of deepfake manipulation techniques. CNNs are essential for analyzing individual frames and extracting spatial features, while RNNs allow the model to capture temporal discrepancies between the frames of a video. This combined architecture allows our model to deeply analyze videos, picking up on subtle inconsistencies that discern deepfake videos from authentic ones, thereby enhancing detection accuracy.

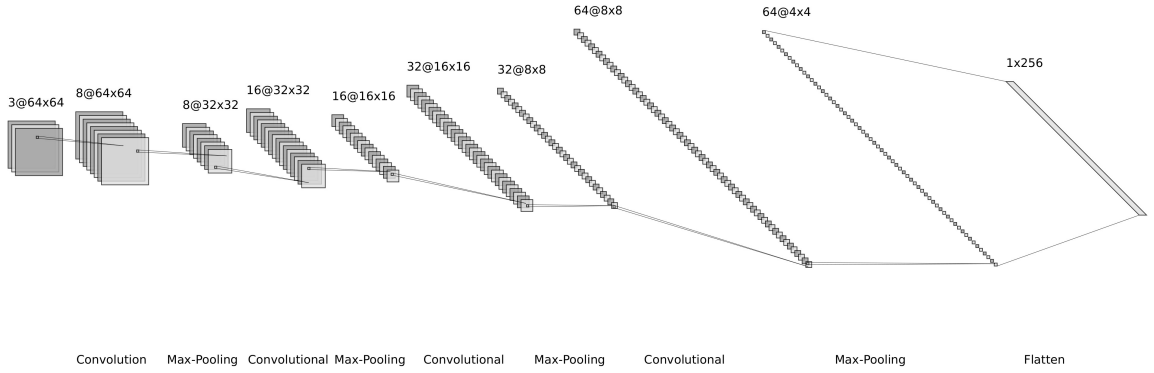



Figure 3.7: Layers of the CNN component of the developed model

### Convolutional Neural Network (CNN) Component

The CNN component of the model is primarily responsible for extracting features from the input data by examining the spatial elements making up the frames. It analyzes the video frames independently, recognizing and encoding any important features, such as edges, textures, or patterns within the input frames. This allows the model to learn visual cues and patterns that characterize real and deepfake videos.

The CNN model is composed of a series of sequential convolutional layers with increasing numbers of filters to capture both abstract and complex spatial patterns. Additionally, subsequent convolutional layers are separated by pooling layers to reduce the spatial dimensions of the feature maps, hence increasing computational efficiency and controlling overfitting. The CNN model architecture can be seen in Figure 3.7 and its layers are defined as follows:

1. **Conv2D Layer (8 filters, kernel size $3 \times 3$, ReLU activation):** This layer takes a stream of input frames, where each frame has dimensions of $64 \times 64$ across three channels (RGB), and applies the Rectified Linear Unit (ReLU) activation function to the feature maps extracted by the filters to introduce non-linearity.

2. **MaxPooling2D Layer (pool size $2 \times 2$):** This layer applies max-pooling to the output of the preceding convolutional layer, reducing the spatial dimensions of the feature maps by selecting the maximum value within each $2 \times 2$ region.

3. **Conv2D Layer (16 filters, kernel size $3 \times 3$, ReLU activation):** This layer applies 16 filters, each with a kernel size of $3 \times 3$, to the reduced feature maps and applies the ReLU activation function to the output.

4. **MaxPooling2D Layer (pool size $2 \times 2$):** Another max-pooling layer that further reduces the spatial dimensions of the convolutional layer output by a factor of 2.

5. **Conv2D Layer (32 filters, kernel size $3 \times 3$, ReLU activation):** This layer applies 32 filters of size $3 \times 3$ to the input feature maps and, similarly, applies the ReLU activation function to the output.

6. **MaxPooling2D Layer (pool size $2 \times 2$):** Another $2 \times 2$ max-pooling layer that further reduces the spatial dimensions of the feature maps produced by the previous convolutional layer.

7. **Conv2D Layer (32 filters, kernel size $3 \times 3$, ReLU activation):** The final convolutional layer applies 64 filters to the input data with a size of $3 \times 3$ followed by the ReLU activation function on the output feature maps.

8. **MaxPooling2D Layer (pool size $2 \times 2$):** The final pooling layer further reduces the spatial dimensions of the feature maps by performing $2 \times 2$ max-pooling.

9. **Flatten():** This additional layer flattens the output of the preceding layers into a 1-dimensional vector suitable to be input to the next component of the model.

The first layer of our main model is a Time-Distributed layer that applies the described CNN model to each frame of the input video stream producing a sequence of flattened feature vectors that are then fed through the RNN component of the model to analyze temporal variations between the extracted feature patterns.

### Recurrent Neural Network (RNN) Component

Despite many advancements in deepfake generation techniques, the majority of synthesized videos often exhibit temporal inconsistencies such as unnatural facial expressions, erratic movements, or discrepancies in lip synchronization over time. Therefore, incorporating an RNN layer in our model enables it to capture and exploit unnatural pattern variations between frames such as abrupt changes in facial features or inconsistencies in motion continuity.

A Long Short-Term Memory (LSTM) layer serves as the main RNN component of our model. The LSTM layer takes the vector of spatial features extracted from the video frames by the CNN component as input and analyzes consecutive frames to capture long-term dependencies between the sequential data, uncovering subtle patterns and temporal dynamics between frames that may indicate the presence of deepfake manipulation.

Our model uses an LSTM layer with 32 units, denoting the dimensionality of the output space, that takes an input sequence of shape (50, 256), as produced by the preceding Time-Distributed layer, where 50 is the number of frames in each video sequence and 256 is the dimension of the flattened vector produced by the CNN component.

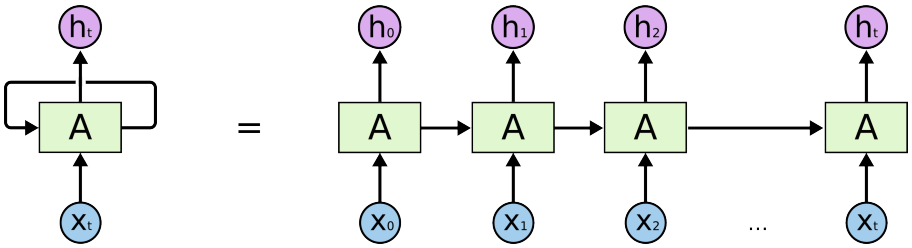



Figure 3.8: The Architecture of a Recurrent Neural Network [31]

At each time step, a sequence of features corresponding to one frame of the input video is fed to the LSTM layer. It processes these features along with the hidden state from the previous time step, as seen in Figure 3.8, capturing any variations between them and updating its internal state accordingly. This process is repeated for each frame in the sequence, allowing the LSTM layer to capture temporal dependencies and patterns across the entire video sequence.

The output of the LSTM layer is a vector of dimensionality 32 representing the model's understanding of the input features' temporal evolution. This vector is then fed to the subsequent layers of the model for further processing and classification.

### Fully Connected Layer

The following layer of the model is a fully connected layer composed of a dense layer along with a dropout function to avoid overfitting.

1. The 16-unit dense layer receives the output of the LSTM layer as input and applies the ReLU activation function to it. This introduces non-linearity to the network, transforming the data into higher-level representations and enabling it to learn complex relationships between the features.

2. A dropout layer with a dropout rate of 0.2 is applied to prevent overfitting by randomly setting 20% of the output to zero during training. This helps the model become more robust by preventing it from memorizing the training data and enabling it to generalize well to unseen data.

### Output Layer

The final layer of the model is a single dense layer with 1 unit that takes the output of the fully connected layer as input and performs binary classification of the videos by applying the Sigmoid activation function. A sigmoid function is used to predict the probability of the input sequence belonging to one of the two classes by mapping the output to a value within the range [0, 1]. If the output value is lower than 0.5, the input video is classified as 'Real' (0); otherwise, it is classified as 'Fake' (1). The closer the output value is to one

end of the range, the higher the predicted probability of it belonging to the corresponding class.

The model is compiled using the Adaptive Moment Estimation (ADAM) optimization algorithm with a learning rate of 0.0001. We selected a Binary cross-entropy loss function as it is the most suitable for binary classification tasks, such as detecting deepfake videos.

### 3.5.2 Data Preparation and Labeling

Before feeding the pre-processed data to the created model, the datasets are compiled into appropriate input shapes suitable for the model layers to process. The training, testing, and validation datasets are set up by calling a function defined to generate a TensorFlow dataset from a given list of video files and their corresponding labels.

#### Compiling Frames

For each dataset, the list of file names designated to it in the data splitting step described in Section 3.4.1 is passed to the function, which iterates through the file names, loading and pre-processing each video file by applying the aforementioned procedures explained in Section 3.4. For every video, an array of frames, encoded as normalized pixel values, is obtained and appended to the list of frames belonging to the dataset.

#### Generating Labels

For the DFDC data, the metadata.json file is first read and loaded into a dictionary composed of each video name along with its corresponding label ('REAL' or 'FAKE'). Compiling a dictionary of video labels makes it easier to label the data streams when generating the input datasets by directly looking up the file name in the dictionary and obtaining its corresponding label, rather than looping through the metadata file for every video. The labels are encoded as binary integers, with 'REAL' videos being labeled (0) and 'FAKE' videos being labeled (1), and appended to the list of labels for the dataset.

Alternatively, for the Celeb-DF data, the files are already sorted into 'Celeb-real' and 'Celeb-synthesis' folders. Therefore, while generating the dataset, the function iterates through each folder separately loading videos whose names are listed in the dataset files and appending the encoded label for each video based on the directory from which it is loaded (0 for videos in the 'Celeb-real' folder and 1 for those in the 'Celeb-synthesis' folder).

#### Creating Datasets

Once the lists of encoded frames and their corresponding labels are collected, a TensorFlow dataset is created by passing the frames and labels to the `Dataset.from_tensor_slices()`

function in the TensorFlow data library. This function returns a dataset with elements corresponding to each pair of frame array and label. If the dataset being generated is for training, it is shuffled with a buffer size of 100 to randomize the order of samples analyzed during training. Finally, the dataset is split into batches of size 32 to be fed to the model sequentially.

#### Testing Data

To evaluate and compare the model's performance on different emotion classes, the testing data attributed for each dataset as explained in Section 3.4.1, is further divided based on the emotion labels extracted from the FER model. For each dataset, the corresponding CSV file generated during the emotion extraction process is loaded and scanned to identify the major emotion label assigned to each video in the testing set by looking up the file names. 6 subsets are compiled for each dataset corresponding to the 6 emotion labels (happiness, sadness, anger, fear, disgust, and surprise) and each testing video is allocated to the appropriate dataset to be used in testing the model.

# Chapter 4

# Results & Limitations

In this chapter, we discuss and interpret the results obtained throughout the study. Section 4.1 provides a comprehensive analysis of the emotions detected in the Celeb-DF and DFDC datasets, highlighting their distribution and prevalence as determined by the emotion extraction process and providing a statistical analysis of the results. Section 4.2 details the experimental setup, including the tools, datasets, and training parameters used for the deepfake detection model. Finally, Section 4.3 outlines the model's performance during training and validation, and its efficiency in detecting deepfake videos when tested across different emotion classes, analyzing how expressed emotions impact the model's performance. The implications of the model's performance and a discussion of the findings are presented in Section 4.4. The chapter also highlights substantial limitations encountered throughout the study in Section 4.5.

## 4.1 Emotion Distribution

In this section, we provide a detailed analysis of the emotions detected in each dataset, highlighting the distribution and prevalence of different emotions. Emotion extraction was performed on both datasets using the FER model outlined in Section 3.3.2. Statistical data analysis was performed on the results of the extraction process to further explore the nature of the emotions portrayed in either dataset as discussed below.

### 4.1.1 Celeb-DF

Running the FER model [12] on the reduced Celeb-DF dataset resulted in the distribution of emotions shown in Figure 4.1. The reduced dataset initially contained a total of 3,114 videos; however, videos displaying no significant emotions were disregarded, leaving 3,065 videos. The number of videos belonging to each emotion class can be seen in Table 4.1. Fear was the most abundant emotion in the dataset, followed by happiness, anger, disgust, sadness, and surprise as shown in Figure 4.1.

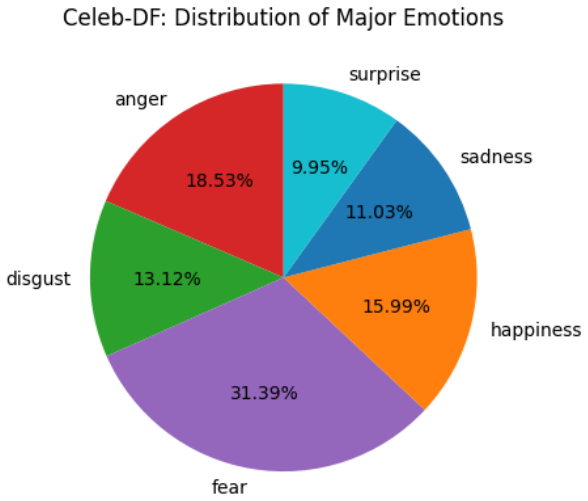
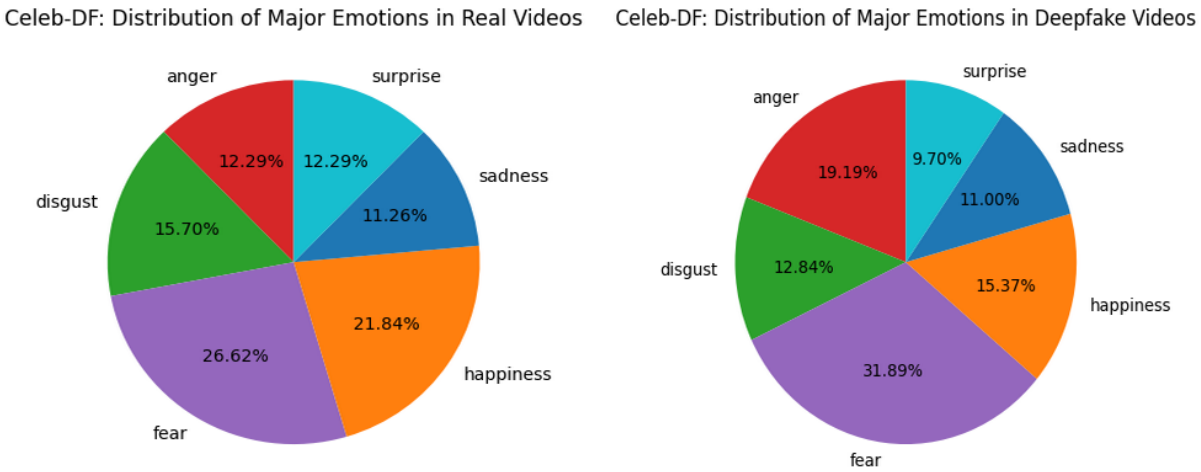



Figure 4.1: Distribution of Major Emotions across the Celeb-DF dataset

| Major Emotion | Label | Count |
|---|---|---|
| anger | FAKE | 532 |
| | REAL | 36 |
| disgust | FAKE | 356 |
| | REAL | 46 |
| fear | FAKE | 884 |
| | REAL | 78 |
| happiness | FAKE | 426 |
| | REAL | 64 |
| sadness | FAKE | 305 |
| | REAL | 33 |
| surprise | FAKE | 269 |
| | REAL | 36 |

Table 4.1: Number of real and deepfake videos from Celeb-DF in each emotion class

To further understand the role emotions play in deepfake detection, the distribution of emotions across real and fake videos were observed individually to compare the variations between them as shown in Figure 4.2.




(a) Real Videos

(b) Deepfake Videos

Figure 4.2: Emotion Distribution across Real (a) and Deepfake (b) Videos in Celeb-DF

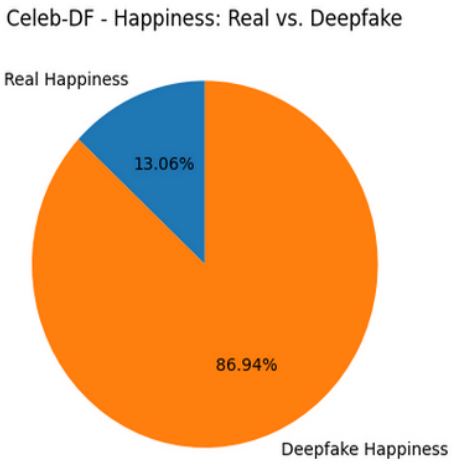
Ideally, the emotions should be distributed across the real and deepfake videos with similar percentages; however, as seen in the figures, some emotions were more prevalent in

the deepfake subset than in the real one and vice versa. For instance, a higher percentage of deepfake videos were detected as portraying fear than the percentage of real videos. Alternatively, the percentage of deepfake videos expressing happiness is seemingly not as high as their real counterparts, indicating that happy expressions might not be as easily synthesized as other emotions.

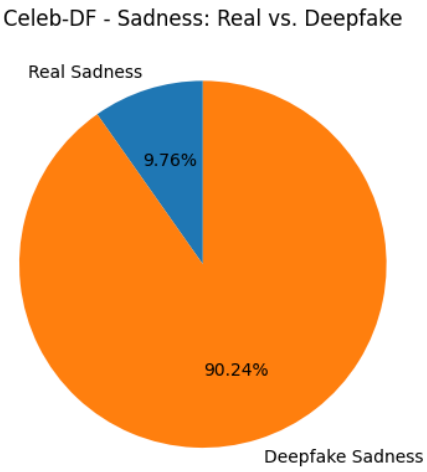
In Figure 4.3, each emotion is individually broken down to show the distribution of real and fake videos across the detected emotion. Comparing each emotion distribution with the general distribution of real and deepfake videos across the dataset shown in Figure 3.4a, there are subtle but noticeable variations in the ratios of deepfake to real videos. The dataset is comprised of 90.53% deepfake videos and 9.47% real videos. Analyzing each individual emotion, we can observe the following trends for this dataset:

- **Happiness:** the ratio of deepfake happiness to real happiness is lower than the general distribution of the dataset, indicating that deepfake videos don't portray happiness as well as other emotions.
- **Sadness:** the ratio of deepfake sadness to real sadness is almost equivalent to the percentages of the dataset, indicating that sadness is the most accurately synthesized emotion.
- **Anger:** the percentage of deepfake videos detected expressing anger is significantly larger than the original distribution of the dataset. This indicates that attempts at synthesizing other emotions may inaccurately be detected as portraying anger.
- **Fear:** similar to anger, the percentage of deepfake videos predicted to be expressing fear is slightly higher than the original distribution of the dataset, signifying that synthesized expressions may inaccurately portray expressions correlated with fear.
- **Disgust:** the ratio of deepfake videos to real videos detected expressing disgust is slightly lower than the original distribution of the dataset, indicating that facial expressions associated with disgust are not as accurately synthesized.
- **Surprise:** the ratio of deepfake videos to real videos detected expressing surprise is slightly lower than the original distribution of the dataset, indicating that deepfake generation techniques cannot accurately synthesize expressions indicative of surprise as well as other emotions.

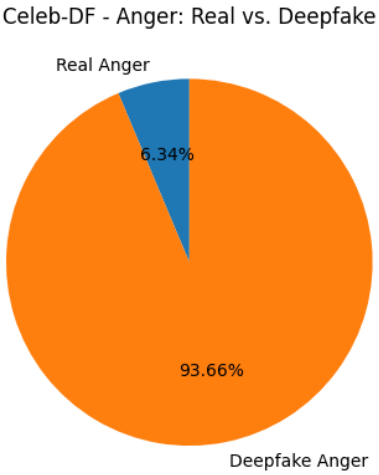
These comparisons indicate that deepfake generation techniques result in facial expressions exhibiting emotions that may vary from the originally portrayed expressions. Sadness is seemingly the most accurately synthesized emotion, while happiness appears to be the most difficult emotion to mimic. Alternatively, failed attempts at replicating true emotions are perceived as portraying either anger or fear.



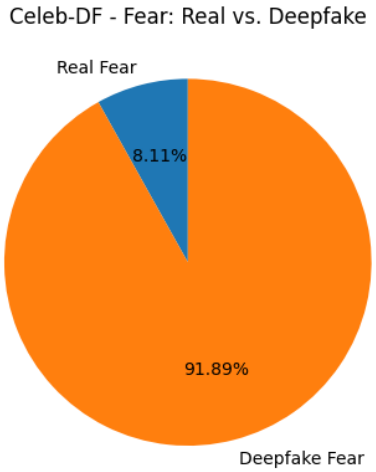

(a) Happiness



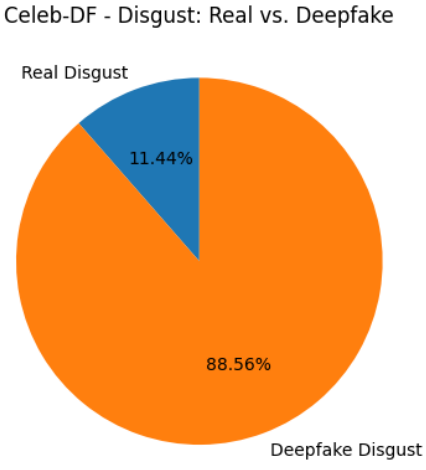

(b) Sadness



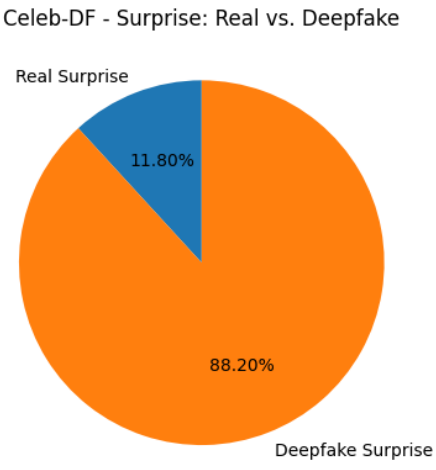

(c) Anger




(d) Fear




(e) Disgust




(f) Surprise

Figure 4.3: Distribution of Real vs. Deepfake videos in Celeb-DF for each emotion

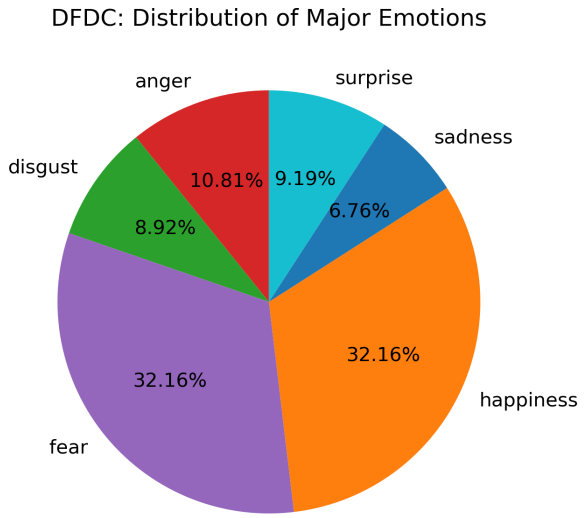



Figure 4.4: Distribution of Major Emotions across the DFDC dataset

| Major Emotion | Label | Count |
|---|---|---|
| anger | FAKE | 29 |
| | REAL | 11 |
| disgust | FAKE | 26 |
| | REAL | 7 |
| fear | FAKE | 100 |
| | REAL | 19 |
| happiness | FAKE | 97 |
| | REAL | 22 |
| sadness | FAKE | 19 |
| | REAL | 6 |
| surprise | FAKE | 25 |
| | REAL | 9 |

Table 4.2: Number of real and deepfake videos from DFDC in each emotion class

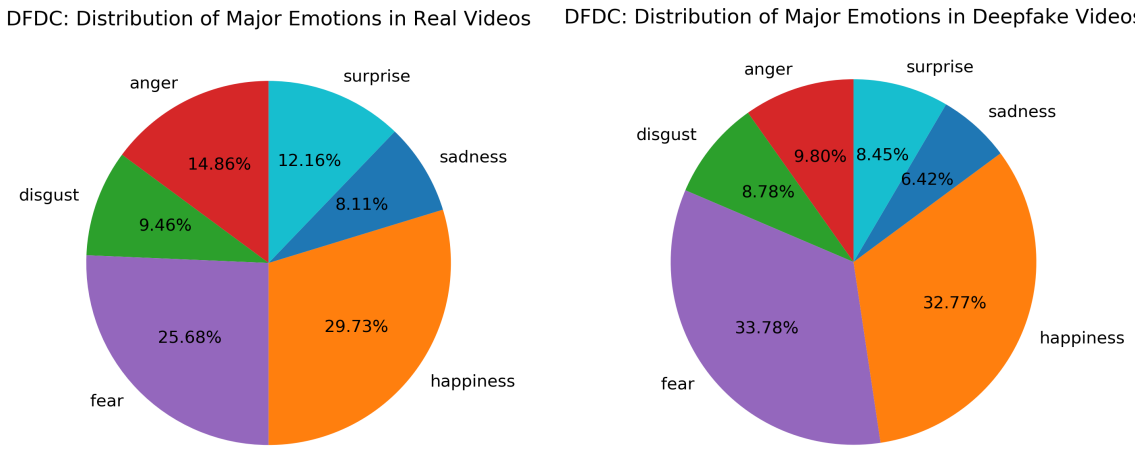
### 4.1.2 DeepFake Detection Challenge (DFDC)

Similarly, the emotions obtained from running the FER model [12] on the DFDC dataset were distributed as shown in Figure 4.4. The DFDC dataset consists of a total of 370 videos, of which 74 are real and 296 are fake, that are distributed among the emotion classes as outlined in Table 4.2. Both fear and happiness are equally the most prevalent emotions in the dataset, followed by anger, surprise, disgust, then sadness as seen in Figure 4.4.

The distributions of emotions across real and deepfake videos were observed individually to compare the variations between them throughout the DFDC dataset as shown in Figure 4.5. This provides an insight into the extent to which emotions in the DFDC dataset can be accurately synthesized.

The figures indicate that some emotions were more prominent in the deepfake subset than in the real one, while others were substantially reduced in the deepfake subset. For instance, the percentage of deepfake videos detected to portray either fear or happiness was significantly higher than the videos exhibiting the same emotions in the real subset. Alternatively, the prevalence of the remaining four emotions in the deepfake subset is significantly less than their existence in the real subset. This variance indicates that not all emotions are synthesized with the same accuracy during deepfake generation.

In Figure 4.6, each emotion is individually analyzed to show the distribution of real and fake videos across the detected emotion. Comparing each emotion distribution with the general distribution of real and deepfake videos across the DFDC dataset as shown in Figure 3.4b, there are a few noticeable variations in the ratios of deepfake to real videos.



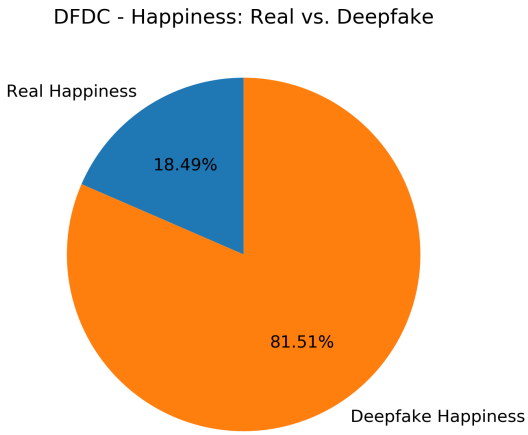

(a) Real Videos

(b) Deepfake Videos

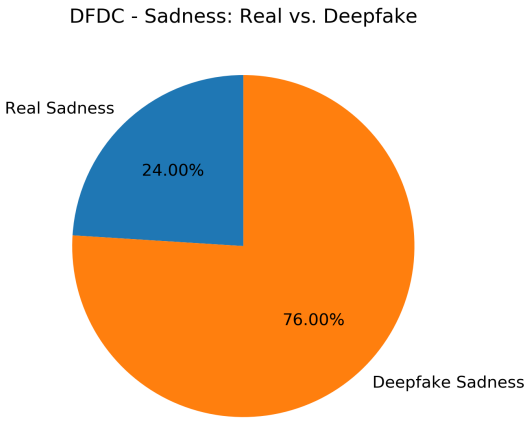
Figure 4.5: Emotion Distribution across Real (a) and Deepfake (b) Videos in DFDC

The dataset is generally comprised of 80.75% deepfake videos and 19.25% real videos. Analyzing each individual emotion, we can observe the following trends for this dataset:

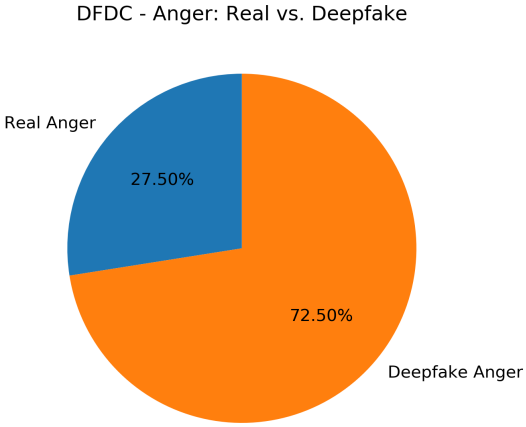
- **Happiness:** the ratio of deepfake happiness to real happiness is negligibly higher than the general distribution of the dataset, indicating that, while some synthesized videos are inaccurately detected as displaying happiness, the majority of the deepfake videos correctly simulate happiness.
- **Sadness:** the percentage of deepfake videos portraying sadness is reduced, indicating that sadness is not as easily mimicked throughout the DFDC dataset.
- **Anger:** the number of deepfake videos detected expressing anger is significantly lower, suggesting that anger is not easily detectable in the synthesized videos of the DFDC dataset.
- **Fear:** similar to happiness, the percentage of deepfake videos predicted to be expressing fear is slightly higher than the original distribution of the dataset, signifying that synthesized expressions may inaccurately portray expressions correlated with fear.
- **Disgust:** the ratio of deepfake videos to real videos detected expressing disgust is slightly lower than the original distribution of the dataset, indicating that facial expressions associated with disgust are not as accurately synthesized.
- **Surprise:** the ratio of deepfake videos to real videos detected expressing surprise is significantly lower than the original distribution of the dataset, indicating that deepfake generation techniques used to create the DFDC dataset fail to accurately synthesize expressions indicative of surprise as well as other emotions.



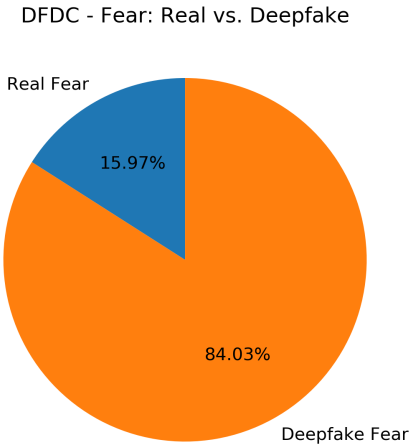

(a) Happiness



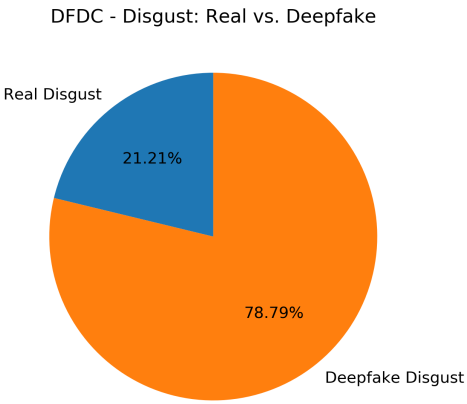

(b) Sadness



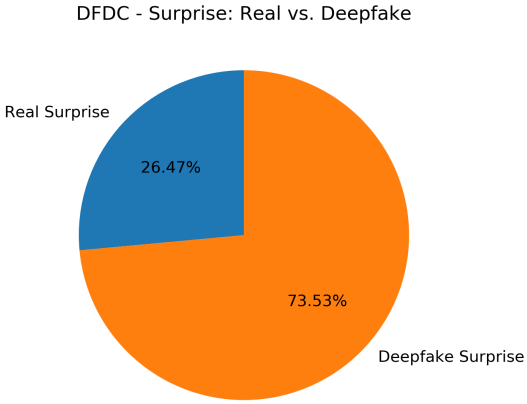

(c) Anger




(d) Fear




(e) Disgust




(f) Surprise

Figure 4.6: Distribution of Real vs. Deepfake videos in DFDC for each emotion

These comparisons indicate that deepfake generation techniques used to compile the DFDC dataset result in facial expressions exhibiting emotions that may vary from the originally portrayed expressions. For this dataset, happiness is seemingly the most accurately synthesized emotion, while anger and surprise appear to be the most difficult emotions to mimic. Alternatively, failed attempts at replicating true emotions are inaccurately detected as portraying fear.

## 4.2 Experiments Setup

This section discusses the tools used to conduct this study and the general setup of the training and evaluation process. The training and evaluation process for our deepfake detection model was conducted on Kaggle, a popular online platform that provides resources and tools for data science and machine learning projects. Kaggle also hosts a wide range of datasets and models to be used directly through the platform. Both the Celeb-DF dataset [36] and the DFDC dataset [6] used in this study were accessed directly through Kaggle.

The videos in each dataset were processed and divided as explained in Sections 3.3 and 3.4 and input to the model. For each dataset, the model was trained on the training data extracted and compiled in Sections 3.5.2 for 15 epochs on batches of size 32. The hyper-parameters of the training process are detailed in Table 4.3.

Table 4.3: Hyperparameters defined for training the model

| Hyperparameter | Value/Type |
|---|---|
| Number of Epochs | 15 |
| Batch Size | 32 |
| Learning Rate | 0.0001 |
| Optimizer | ADAM |
| Activation Function | Sigmoid |
| Loss Function | Binary Cross-entropy |

## 4.3 Results

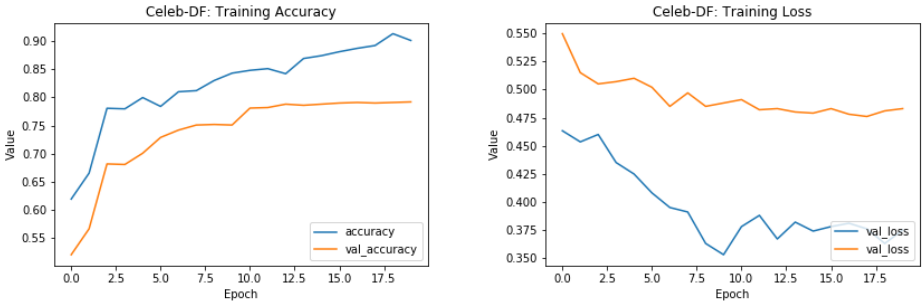
The following section discusses and analyzes the model's performance on both the Celeb-DF and DFDC datasets. First, we examine the performance of the model during the training and validation phase on each of the two datasets in Section 4.3.1. Then, in Section 4.3.2, we observe the model's efficiency at detecting deepfake videos belonging to each dataset by analyzing its performance on different emotion classes and interpreting how expressed emotions affect its performance.

### 4.3.1 Training and Validation Performance

To evaluate our model's performance and robustness, we examine the training and validation performance of the model on both datasets: Celeb-DF and DFDC. We present detailed metrics, including accuracy and loss, observed over multiple epochs to illustrate how the model's performance evolved during training. Additionally, the differences in validation accuracy and loss are analyzed to assess the model's generalization capability across both datasets.

#### Celeb-DF

Training the model on the Celeb-DF dataset yielded the metrics displayed in Figure 4.7. Figure 4.7a plots the training accuracy and the validation accuracy over multiple epochs. Alternatively, Figure 4.7b shows the curves representing the training loss and validation loss values throughout the training process.




(a) Accuracy Curves (b) Loss Curves

Figure 4.7: Training and Validation metrics on the Celeb-DF dataset

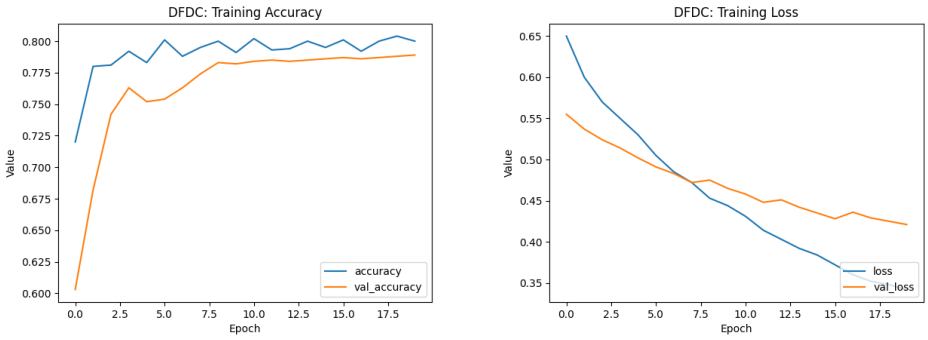
Examining the accuracy curve, it is evident that both accuracy metrics experience a sudden increase over the first 2 epochs, and then continue to rise gradually for the rest of the training process.

Figure 4.7b shows that the training and validation losses experience a subtle but generally decreasing trend across the epochs.

#### DeepFake Detection Challenge (DFDC)

When trained on the DFDC dataset the model resulted in the metrics illustrated in Figure 4.8. The figure depicts the progression of training and validation accuracy and loss across the training epochs.

Figure 4.8a shows a significant increase in the training and validation accuracy of the model over the first few epochs. The training accuracy fluctuates throughout the rest of




(a) Accuracy Curves (b) Loss Curves

Figure 4.8: Training and Validation metrics on the DFDC dataset

the training process while still maintaining a general increasing trend. Contrarily, the validation accuracy stabilizes around halfway through the training process, indicating the model's ability to effectively generalize to unseen data.

In Figure 4.8b, both the training and validation loss values decrease steadily throughout the training process, experiencing very minimal fluctuations.

### 4.3.2 Testing Performance

After individually training the model on both datasets, its performance on each dataset is evaluated using the emotion-labeled testing datasets compiled in Section 3.5.2. For each dataset, we examine the test loss and accuracy of the model when it is run on each emotion class. Additionally, confusion matrices for each emotion are presented providing insight into the model's prediction capabilities by showcasing the distribution of true and false classifications of the test videos.

## Celeb-DF

The results of evaluating the model's performance on the categorized testing subsets compiled from the Celeb-DF dataset are detailed below, separately discussing the metrics obtained by testing each emotion class.

#### Happiness

The testing dataset contained 102 videos exhibiting happiness. Figure 4.9 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying happiness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.4.

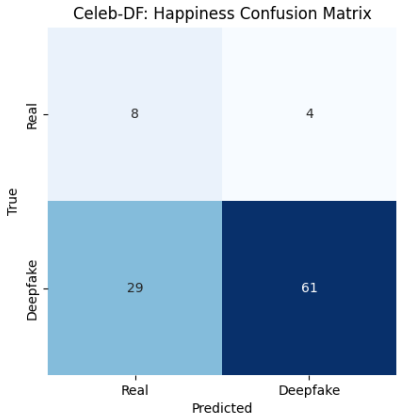



Figure 4.9: Happiness Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.6765 |
| Loss | 0.4872 |
| Precision | 0.9385 |
| Recall | 0.6777 |

Table 4.4: Happiness Testing Metrics

### Sadness

A total of 61 videos in the testing dataset were detected as portraying sadness. Figure 4.10 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying sadness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.5.

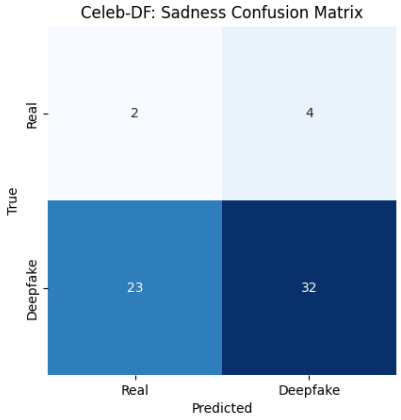



Figure 4.10: Sadness Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.5773 |
| Loss | 0.4912 |
| Precision | 0.8888 |
| Recall | 0.5818 |

Table 4.5: Sadness Testing Metrics

### Anger

The testing dataset contained 120 videos portraying anger. Figure 4.11 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying anger. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.6.

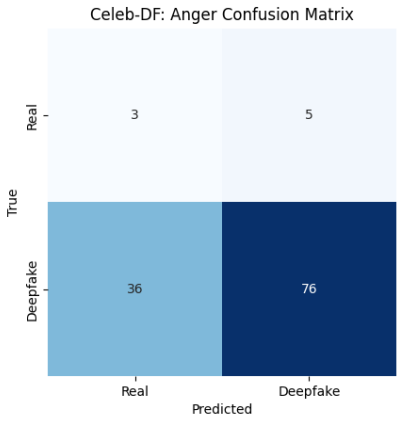



Figure 4.11: Anger Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.6583 |
| Loss | 0.4901 |
| Precision | 0.9382 |
| Recall | 0.6785 |

Table 4.6: Anger Testing Metrics

### Fear

Fear was the most prevalent emotion in the testing dataset with 196 videos. Figure 4.12 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying fear. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.7.

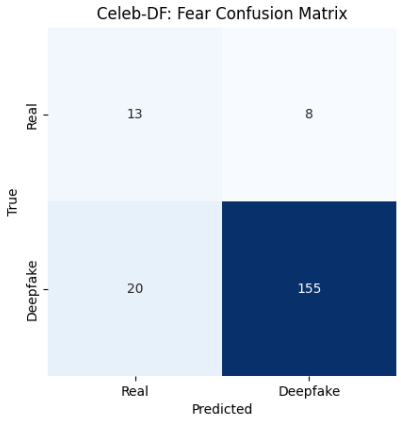



Figure 4.12: Fear Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.8571 |
| Loss | 0.3487 |
| Precision | 0.9509 |
| Recall | 0.8857 |

Table 4.7: Fear Testing Metrics

### Disgust

The testing dataset contained 73 videos expressing disgust. Figure 4.13 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying disgust. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.8.

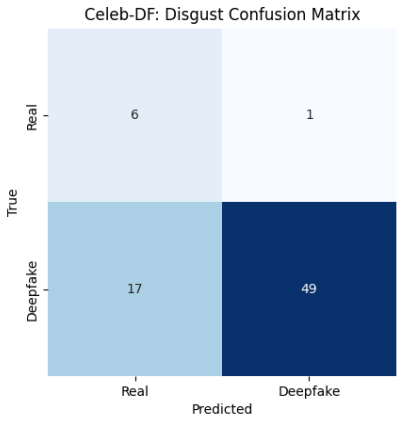



Figure 4.13: Disgust Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.7534 |
| Loss | 0.3921 |
| Precision | 0.9800 |
| Recall | 0.7424 |

Table 4.8: Disgust Testing Metrics

### Surprise

The remaining 61 videos in the testing dataset were categorized into the Surprise emotion class. Figure 4.14 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying sadness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.9.

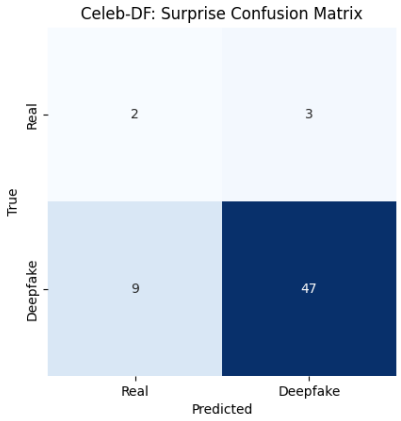



Figure 4.14: Surprise Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.8032 |
| Loss | 0.3702 |
| Precision | 0.9400 |
| Recall | 0.8392 |

Table 4.9: Surprise Testing Metrics

## DeepFake Detection Challenge (DFDC)

The results of evaluating the model's performance on the categorized testing subsets compiled from the DFDC dataset are detailed below, separately discussing the metrics obtained by testing each emotion class.

### Happiness

The testing dataset was found to contain 27 videos exhibiting happiness. Figure 4.15 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying happiness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.10.

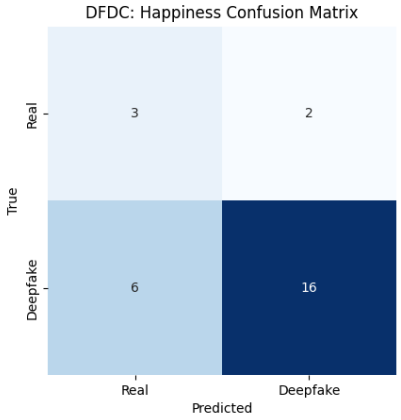



Figure 4.15: Happiness Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.7037 |
| Loss | 0.1801 |
| Precision | 0.8889 |
| Recall | 0.7272 |

Table 4.10: Happiness Testing Metrics

### Sadness

7 of the videos in the testing dataset were detected as portraying sadness. Figure 4.16 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying sadness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.11.

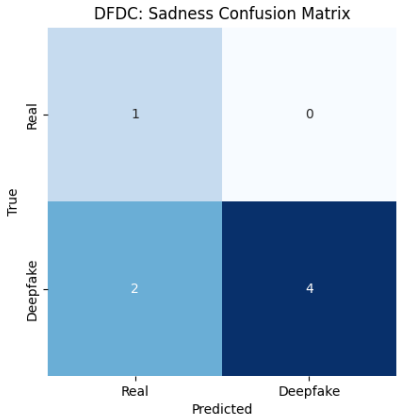



Figure 4.16: Sadness Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.7143 |
| Loss | 0.1582 |
| Precision | 1.0000 |
| Recall | 0.6667 |

Table 4.11: Sadness Testing Metrics

### Anger

The testing dataset contained 6 videos portraying anger. Figure 4.17 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying anger. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.12.

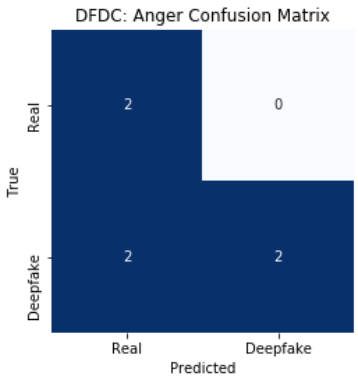



Figure 4.17: Anger Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.6667 |
| Loss | 0.2268 |
| Precision | 1.0000 |
| Recall | 0.5000 |

Table 4.12: Anger Testing Metrics

### Fear

Fear was the second most prevalent emotion in the testing dataset with 20 videos belonging to this class. Figure 4.18 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying fear. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.13.

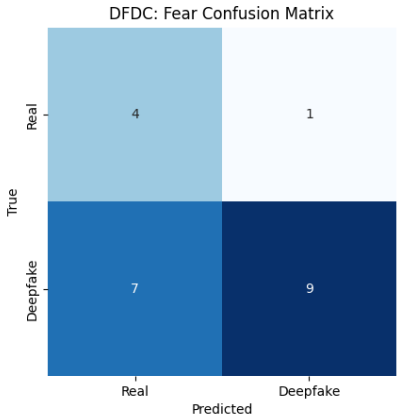



Figure 4.18: Fear Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.6500 |
| Loss | 0.2390 |
| Precision | 0.9000 |
| Recall | 0.5625 |

Table 4.13: Fear Testing Metrics

### Disgust

The testing dataset contained 6 videos expressing disgust. Figure 4.19 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying disgust. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.14.

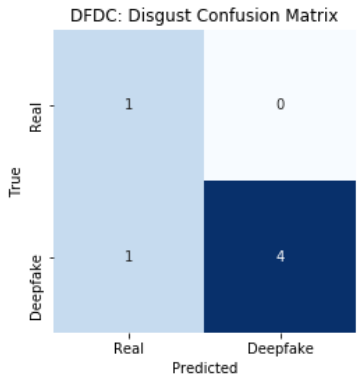



Figure 4.19: Disgust Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.8333 |
| Loss | 0.1518 |
| Precision | 1.0000 |
| Recall | 0.8000 |

Table 4.14: Disgust Testing Metrics

### Surprise

The remaining 9 videos in the testing dataset were categorized into the Surprise emotion class. Figure 4.20 shows the confusion matrix detailing the distribution of predicted labels obtained from the model compared to the true labels of the test videos conveying sadness. The performance metrics obtained by analyzing the confusion matrix, as well as the test loss evaluated by the model are summarized in Table 4.15.

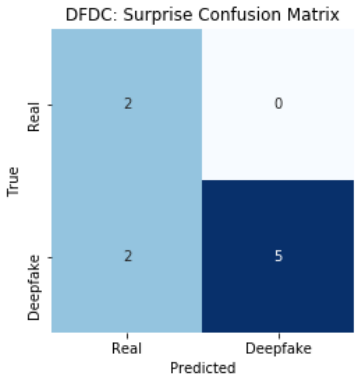



Figure 4.20: Surprise Confusion Matrix

| Metric | Value |
|---|---|
| Accuracy | 0.7778 |
| Loss | 0.1768 |
| Precision | 1.0000 |
| Recall | 0.7143 |

Table 4.15: Surprise Testing Metrics

## 4.4 Results Analysis and Discussion

Careful analysis of the training and validation performance on the Celeb-DF and DFDC datasets reveals that the model learns effectively in the initial epochs and continues to improve, albeit more gradually, as training progresses. The steady decrease in loss values across both datasets underscores the model's ability to minimize errors during training.

When evaluating the model's performance on emotion-labeled testing datasets, it is evident that the model's effectiveness varies across different emotions for both datasets.

Analyzing the testing metrics exhibited by the Celeb-DF dataset, the model demonstrated moderate accuracy, with varying performance depending on the specific emotion being detected. It achieved the highest accuracy for videos portraying fear (0.8571), indicating that fear expressions have distinctive features that the model can effectively learn and identify. Contrarily, the model struggled more with sadness (accuracy: 0.5773), suggesting that detecting the subtler facial cues associated with this emotion posed a greater challenge. Other emotions, such as happiness, anger, disgust, and surprise, yielded intermediate accuracy values ranging from approximately 0.6583 to 0.8032. These results highlight the model's variable sensitivity to different emotional expressions, reflecting the complexity and diversity of human facial emotions. Precision values were generally high across all emotions, but recall values varied, indicating that while the model is good at avoiding false positives, it sometimes fails to detect all true positives, particularly for emotions with less pronounced expressions.

Upon examining the testing metrics of the DFDC dataset, the precision values for most emotion classes were consistently high, indicating that the model is good at correctly identifying real videos in this dataset, while the variations in recall suggest that certain emotions, such as anger and fear, pose more challenges for the model as they achieve lower accuracies of 0.6667 and 0.6500. These variations can be attributed to the nature of these emotions as anger and fear expressions emotions like anger and fear often involve complex and dynamic facial expressions, such as furrowed brows, wide eyes, and dropped jaws.

Table 4.16: Testing accuracies achieved for each emotion class

| | Celeb-DF | DFDC |
|---|---|---|
| Happiness | 67.65% | 70.37% |
| Sadness | 57.73% | 71.43% |
| Anger | 65.83% | 66.67% |
| Fear | **85.71**% | 65.00% |
| Disgust | 75.34% | **83.33%** |
| Surprise | 80.32% | 77.78% |

The rapid changes in facial muscle movements associated with these emotions can make it challenging for the model to distinguish between genuine and manipulated content. It should be noted that the variance in metrics could also be attributed to the limited size of the dataset sample which most likely results in skewed predictions.

The overall results and accuracy values achieved by the model for each emotion class in either dataset are summarized in Table 4.16, highlighting the maximum accuracy for each dataset. These insights highlight the need for further refinement and potential adjustments in the model to improve its robustness across all emotional expressions.

## 4.5 Limitations

Throughout this study, several limitations were encountered that impacted the overall performance and generalizability of the deepfake detection model. A primary challenge was the limited availability of suitable datasets for training, which restricted the scope of our analysis. The datasets we utilized were heavily unbalanced, with a significantly higher number of deepfake videos compared to real ones, which could bias the model's learning process. Specifically, the accessible sample from the DFDC dataset was relatively small, which had an impact on the reliability and validity of the findings. Moreover, the DFDC videos were often of low quality, featuring poor lighting conditions and ambiguous emotional cues, further complicating the detection task. As previously discussed, a major limitation of the study was the lack of significant audio data in datasets, which rendered the initial approach of incorporating Speech Emotion Recognition impractical. Lastly, hardware limitations and computational capabilities necessitated the use of smaller data samples for training and evaluation, potentially constraining the model's capacity to learn from a more comprehensive dataset. These factors collectively highlight the need for more extensive, balanced, and high-quality datasets, as well as improved computational resources, to advance deepfake detection methodologies.

# Chapter 5

# Conclusion & Future Work

## 5.1 Conclusion

This thesis aimed to address the growing challenge of detecting deepfakes by leveraging emotion recognition as a novel approach to develop a deep learning detection model and explore how accurately it performs on different emotion classes to gain a better understanding of the role emotions play in deepfake detection.

The methodology for our approach first involved exploring and evaluating various datasets, particularly the Celeb-DF and DFDC datasets, to train and test our deepfake detection model. Emotion recognition and label extraction were performed using a pre-trained FER model to gain insightful information on the distribution of emotion classes throughout the datasets, Pre-processing steps were applied to the Celeb-DF and DFDC datasets to extract important features from video frames. The datasets were then organized into training, testing, and validation subsets and adjusted to fit the model input for the training process. The testing subsets were further divided by emotion class to assess the model's accuracy for each class. The developed model combined the capabilities of CNNs in image classification and RNNs in analyzing sequential data to meticulously analyze key features in consecutive video frames, exploiting both temporal and spatial analysis.

Training and evaluating our model revealed varying trends across different emotion classes, with more expressive emotions achieving higher accuracy, indicating that emotions with more distinct expressions are easier to detect. In contrast, subtler emotions such as sadness posed greater challenges, resulting in lower accuracy. These findings underscore the significant role that emotions play in deepfake detection, as the discrepancies in emotional cues between real and synthesized videos can serve as key indicators for detection.

In conclusion, the study established that emotions indeed play a crucial role in deepfake detection. Different emotional expressions were detected with varying degrees of

accuracy, highlighting the complexities involved in accurately replicating human emotions and expressions in synthesized deepfake videos. The field of emotion recognition proves to be a promising approach to developing and augmenting deepfake detection models.

## 5.2 Future Work

Future work in this research area could address some of the limitations imposed throughout this study contributing to enhancing the reliability and accuracy of current deepfake detection methodologies.

One crucial area that requires further contribution is expanding and assembling representative datasets incorporating a wide variety of data to ensure the models can generalize effectively across different contexts and scenarios. Future datasets including more diverse and balanced samples will better reflect real-world conditions improving current detection models' robustness and generalization. Additionally. improving the quality of data, particularly in terms of resolution and clarity, is essential. High-quality data allows for more precise feature extraction and better model training, leading to improved performance. Exploring different modalities is also a promising direction within which a lot of work can be conducted. Integrating visual, auditory, and possibly textual data could provide a more holistic approach to detecting deepfakes, leveraging the strengths of each modality to improve overall accuracy. This multimodal approach can capture inconsistencies across various data types, enhancing detection capabilities. Given the novelty of this research area, ongoing research and experimentation have the potential to uncover new innovative approaches and techniques for detecting deepfake videos, staying ahead of increasingly sophisticated manipulation techniques and the threats they impose.

# Appendix

# Appendix A

# Lists

| | |
|---|---|
| **ADAM** | Adaptive Moment Estimation |
| **AI** | Artificial Intelligence |
| **AUC** | Area Under Curve |
| **AWS** | Amazon Web Services |
| **BLSTM** | Bidirectional Long Short-Term Memory |
| **CCC** | Concordance Correlation Coefficient |
| **CK+** | Extended Cohn-Kanade |
| **CNN** | Convolutional Neural Network |
| **CSV** | Comma Separated Values |
| **DFDC** | DeepFake Detection Challenge |
| **DL** | Deep Learning |
| **FER** | Facial Emotion Recognition |
| **GAN** | Generative Adversarial Network |
| **JSON** | JavaScript Object Notation |
| **LLD** | Low-Level Descriptor |
| **LSTM** | Long Short-Term Memory |
| **RNN** | Recurrent Neural Network |
| **ReLU** | Rectified Linear Unit |
| **S3FD** | Single Shot Scale-Invariant Face Detector |
| **SER** | Speech Emotion Recognition |
| **SNR** | Signal-to-Noise Ratio |
| **SSD** | Synthetic Speech Detector |
| **STFT** | Short Time Fourier Transform |

| | |
|---|---|
| **TTS** | Text-to-Speech |
| **VC** | Voice Conversion |

# List of Figures

# List of Tables

# Bibliography


[1] ai greenscreen.

[2] Marwan Ali Albahar and Jameel Almalki. Deepfakes: Threats and countermeasures systematic review. 2019.

[3] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.

[4] Hasan Dag Mehmet Unal Arash Heidari, Nima Jafari Navimipour. Deepfake detection using deep learning methods a systematic and comprehensive review. *WIREs Data Mining and Knowledge Discovery, e1520*, 2023.

[5] AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[6] djdj Irina Kofman JE Tester JLElliott Joshua Metherd Julia Elliott Mozaic Phil Culliton Sohier Dane Woo Kim benpflaum, Brian G. Deepfake detection challenge, 2019.

[7] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[8] Ben Pflaum Nicole Baram Cristian Canton Ferrer Brian Dolhansky, Russ Howes. The deepfake detection challenge (dfdc) preview dataset, 2019.

[9] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[10] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359, December 2008.

[11] Heather Chen and Kathleen Magramo. Finance worker pays out $25 million after video call with deepfake "chief financial officer", Feb 2024.

[12] Madhur Chhajed. Feedback system using facial emotion recognition, Dec 2020.

[13] Emanuele Conti, Davide Salvi, Clara Borrelli, Brian Hosler, Paolo Bestagini, Fabio Antonacci, Augusto Sarti, Matthew Stamm, and Stefano Tubaro. Deepfake speech detection through emotion recognition: A semantic approach. pages 8962–8966, 05 2022.

[14] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset, 2020.

[15] Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 1459–1462, New York, NY, USA, 2010. Association for Computing Machinery.

[16] HaleyyBaylee. Disney princesses. . . . move aside.

[17] Brian Hosler, Davide Salvi, Anthony Murray, Fabio Antonacci, Paolo Bestagini, Stefano Tubaro, and Matthew C. Stamm. Do deepfakes feel emotions? a semantic approach to detecting deepfakes via emotional inconsistencies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1013–1022, June 2021.

[18] Vinaya Sree Katamneni and Ajita Rattani. Mis-avoidd: Modality invariant and specific representation for audio-visual deepfake detection, 2023.

[19] Pavel Korshunov and Sebastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection, 2018.

[20] Prabhat Kumar, Mayank Vatsa, and Richa Singh. Detecting face2face facial reenactment in videos, 2020.

[21] Yuezun Li, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*, Seattle, WA, United States, 2020.

[22] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics, 2020.

[23] A. Lieto, D. Moro, F. Devoti, C. Parera, V. Lipari, P. Bestagini, and S. Tubaro. "hello? who am i talking to?" a shallow cnn approach for human vs. bot speech classification. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2577–2581, 2019.

[24] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 94–101, 2010.

[25] Yoshi-Taka Matsuda, Tomomi Fujimura, Kentaro Katahira, Masato Okada, Kenichi Ueno, Kang Cheng, and Kazuo Okanoya. The implicit processing of categorical and dimensional strategies: An fmri study of facial emotion perception. *Frontiers in human neuroscience*, 7:551, 09 2013.

[26] Gary McKeown, Michel Valstar, Roddy Cowie, Maja Pantic, and Marc Schroder. The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE Transactions on Affective Computing*, 3(1):5–17, 2012.

[27] Amr Megahed, Qi Han, and Sondos Fadl. Exposing deepfake using fusion of deep-learned and hand-crafted features. *Multimedia Tools and Applications*, 83:1–21, 09 2023.

[28] Danae Mercer. Scary new ai filter.

[29] Trisha Mittal, Uttaran Bhattacharya, Rohan Chandra, Aniket Bera, and Dinesh Manocha. Emotions don't lie: An audio-visual deepfake detection method using affective cues. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 2823–2832, New York, NY, USA, 2020. Association for Computing Machinery.

[30] Susie Ruiz-Lichter Nicole Brenner. Why the taylor swift ai scandal is pushing lawmakers to address pornographic deepfakes, Apr 2024.

[31] Christopher Olah. Understanding lstm networks, Aug 2015.

[32] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

[33] Imran Rahman-Jones. Taylor swift deepfakes spark calls in congress for new legislation, Jan 2024.

[34] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *International Conference on Computer Vision (ICCV)*, 2019.

[35] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos, 2020.

[36] Reuben Suju Varghese. Celeb df (v2), Apr 2023.

[37] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I, 2001.

[38] Mika Westerlund. The emergence of deepfake technology: A review. *Technology Innovation Management Review*, 9:40–53, 11/2019 2019.

[39] Junichi Yamagishi, Massimiliano Todisco, Md Sahidullah, Héctor Delgado, Xin Wang, Nicolas Evans, Tomi Kinnunen, Kong Aik Lee, Ville Vestman, and Andreas Nautsch. Asvspoof 2019: The 3rd automatic speaker verification spoofing and countermeasures challenge database, Jun 2019.

[40] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z. Li. $S^3$fd: Single shot scale-invariant face detector, 2017.