

# Deepfake Detection Techniques

*A Project Report submitted by*

**Shivam Sharma**

**M22AI633**

*in partial fulfillment of the requirements for the award of the degree of*

**Master of Technology**

**in**

**Data and Computational Science**



॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

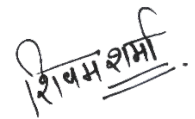
**Indian Institute of Technology Jodhpur**

**Name of the Department**

*December 2023*

## Declaration

I hereby declare that the work presented in this Project Report titled Deepfake Detection Techniques –Master of Technology in Data and Computational Science submitted to the Indian Institute of Technology Jodhpur in partial fulfilment of the requirements for the award of the degree of Master of Technology in Data and Computational Science submitted., is a bonafide record of the research work carried out under the supervision of Professor Dr. Sandeep Kumar Yadav . The contents of this Project Report in full or in parts, have not been submitted to, and will not be submitted by me to, any other Institute or University in India or abroad for the award of any degree or diploma.



**Signature**

*Mr. Shivam Sharma*

M22AI633

## **Certificate**

This is to certify that the Project Report titled Deepfake Detection Techniques, submitted by Mr. Shivam Sharma (M22AI633) to the Indian Institute of Technology Jodhpur for the award of the degree of Master of Technology in Data and Computational Science is a bonafide record of the research work done by him under my supervision. To the best of my knowledge, the contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

### **Signature**

Dr. Sandeep Kumar Yadav  
Department of Electrical Engineering  
Indian Institute of Technology Jodhpur  
Karwar, 342037, India

## Abstract

The Master of Technology in Data and Computational Science Program of study requires each student to undertake research in the chosen area of study and to submit a thesis on it in consultation with the faculty member(s) supervising the same. The Master of Technology in Data and Computational Science Project is included in the curriculum with a view to synthesize the various components of the research work undertaken during the of the Master of Technology in Data and Computational Science Program at IIT Jodhpur. Creating a Project Report document of the research undertaken is part of the skill building training of the student in technical communications. Here, the emphasis is on presenting a technical matter in an objective written form.

This document is a record of the mandatory guidelines to be followed while preparing the of the *Project Report* document to be submitted at the end of the Master of Technology in Data and Computational Science Program. It prescribes typical contents that an Master of Technology in Data and Computational Science Project Report document usually should contain and provides the format of its presentation. While most of these guidelines are prescriptive, some are subjective; but towards ensuring a relatively uniform style of presentation of all Master of Technology in Data and Computational Science Project Report being submitted at the Institute, these subjective guidelines are expected to help in setting at least a reasonable minimum expectation of the presentation level of the work accomplished in the research program.

All students pursuing Master of Technology in Data and Computational Science Program are urged to read the contents and form of this document carefully, and prepare their Project Report document as prescribed. It is hoped that this document will lead to a modest beginning at the Institute towards imparting education in professional written presentations.

# TABLE OF CONTENTS

INTRODUCTION .....6

LITERATURE SURVEY .....8

    2.1 DeepFake Detection by Analyzing Convolutional Traces <sup>[1]</sup> .....8

    2.2 DeepFake Image Detection <sup>[2]</sup> ..... 10

    2.3 Deepfakes Creation and Detection Using Deep Learning <sup>[5]</sup>. .... 12

    2.4 Exploring Depth Information for Face Manipulation Detection <sup>[3]</sup> ..... 14

MODELS AND ALGORITHMS ..... 16

    3.1 Dataset Gathering ..... 16

    3.2 Dataset preparation..... 16

    3.3 Model Creation..... 16

    3.4 Experiments..... 18

RESULTS AND FUTURE SCOPE..... 19

CONCLUSION .....22

REFERENCES .....23

## CHAPTER-1

### INTRODUCTION

Deepfake as the name suggests is blurring the limits of real and fake object attributes like audio, video etc. Deepfakes are proving as dangerous tool as they are extremely hard to detect by eyes, as shown in Figure 1



*Figure 1: Original & Deepfake image*

Deepfakes technique uses the deep learning algorithms to manipulate the audio, video, image etc. of the original content, commonly used techniques to create deepfakes are generative adversarial networks (GAN). In GAN there are two AI/ML algorithms working simultaneously where encoder's task is to make fool the decoder and decoder's task is to identify whether the given output from encoder

is authentic or not, encoder continuously try to improve its performance and whole process then reach to a phase where decoder will not be able to identify the authenticity of the output.

This encoder can now be used for generating the manipulated images after taking authentic image as input. In these days automated tools are easily available that can create deepfakes very effectively, that is why deepfakes are becoming threat for the society as there are cases of voice converting, face swapping etc. increases over time, the recent case came for a Bollywood celebrity's face used in one video without consent, and the video was deepfake so perfectly is was hard to detect authenticity of it just by looking at it

The goal of the project is to explore and apply various deep learning techniques that can predict whether the given image is authentic or deepfake. The GAN model architecture as shown in Figure 2 is an example of deepfake generation technique.

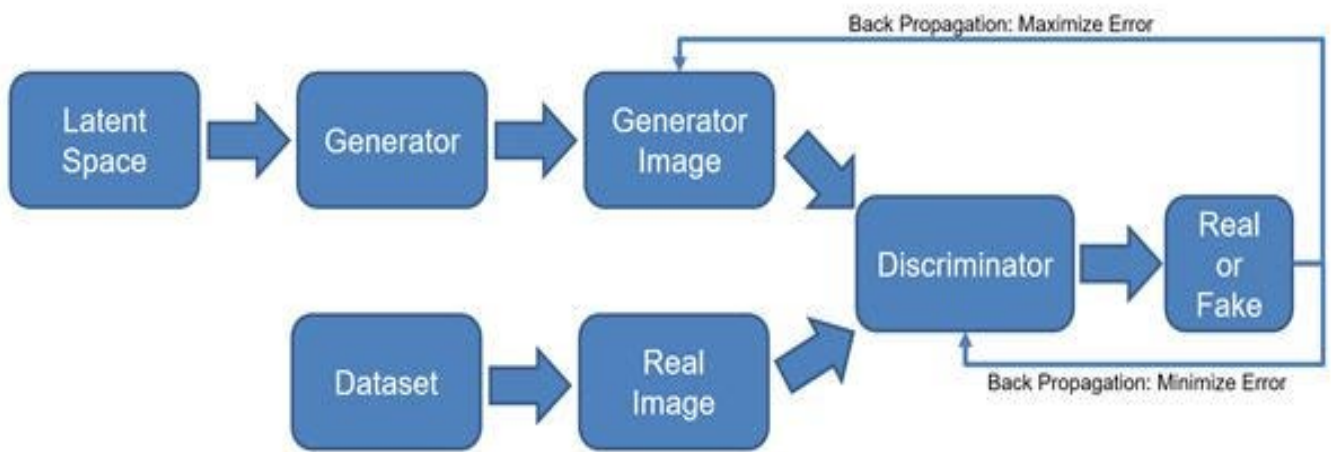


Figure 2: GAN for deepfake generation

Now deepfakes detection techniques can use the visual as well as hidden features of a given image to identify authenticity of the given image, for example while GAN generate the deepfake it can leave traces of the operations used to generate the deepfake, those can be used to identify clues like what was the kernel size used etc. [1], a deepfake can change the person's face depth maps values [2]. in following sections, we discuss related work done for this project, our proposed methodology and results and future scope of the project.

## CHAPTER-2

### LITERATURE SURVEY

#### 2.1 Deepfake Detection by Analyzing Convolutional Traces <sup>[1]</sup>

In this paper the focus is on the images of human faces, here deepfake detection technique was explored for the given image using Expectation Maximization (EM) algorithm. Here they used various datasets generated by different GANs like GDWCT, STARGAN, ATGAN, STYLEGAN, STYLEGAN2 along with the CELEBA dataset that contained only authentic face images of celebrities. They used the EM algorithm to estimate local features of the images. It gave them a vector of features ~k of 24 elements.

After obtaining vector k of features, they applied basic binary classification algorithms as shown in Figure 3

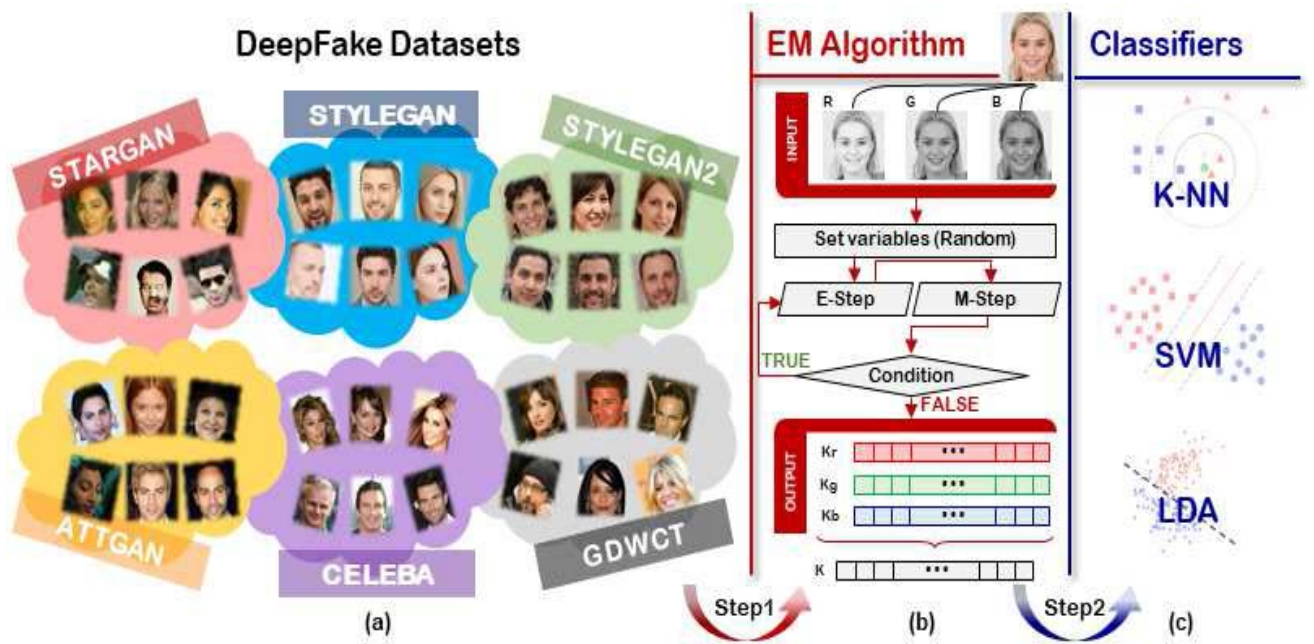


Figure 3: Proposed Methodology

In EM algorithm consist of two steps:

1. Expectation Step
2. Maximization Step

That leads to formula.

$$\sum_{s,t=-\alpha}^{\alpha} k_{s,t} \left( \sum_{x,y} w[x,y] I[x+i, y+j] I[x+s, y+t] \right) =$$

$$= \sum_{x,y} w[x,y] I[x+i, y+j] I[x,y]$$

The EM algorithm is as follows:



---

**Data:** Image  $I$

**Result:**  $\vec{k}$

Initialize  $N$  //Kernel size

Initialize  $\sigma_0$

Set  $\vec{k}$  random of size  $N \times N$

Set  $R, P, W$  matrices with 0 values of the same size as  $I$

Set  $p_0$  as 1/size of the range of values of  $I$

**for**  $n = 1; n < 100$   $n++ = 1$  **do**

    //Expectation Step

**for**  $\forall$  values in  $I$  **do**

$$R[x, y] = \left| I[x, y] - \sum_{s, t = -\alpha}^{\alpha} k_{s, t} I[x + s, y + t] \right|$$

$$P[x, y] = \frac{1}{\sigma_n \sqrt{2\pi}} e^{-\frac{R[x, y]^2}{2\sigma_n^2}}$$

$$W[x, y] = \frac{P[x, y]}{P[x, y] + p_0}$$

    //Maximization Step

    Calculate  $k_{s, t}^{(n+1)}$  as shown in the formula

They were able to attain excellent results after classification, the results are shown below:

	CELEBA Vs ATTGAN				CELEBA Vs GDWCT				CELEBA Vs STARGAN				CELEBA Vs STYLEGAN				CELEBA Vs STYLEGAN2			
	Kernel Size				Kernel Size				Kernel Size				Kernel Size				Kernel Size			
	3x3	4x4	5x5	7x7	3x3	4x4	5x5	7x7	3x3	4x4	5x5	7x7	3x3	4x4	5x5	7x7	3x3	4x4	5x5	7x7
3-NN	92.67	86.50	84.50	85.33	88.40	73.17	73.00	74.33	90.50	89.00	88.67	85.17	93.00	99.65	98.26	99.55	96.99	99.61	98.75	97.77
5-NN	92.00	86.50	84.83	86.17	88.40	75.67	74.17	76.67	88.83	88.83	88.17	85.00	93.00	99.65	98.26	99.32	97.39	99.61	98.21	97.55
7-NN	91.00	87.67	85.33	85.67	88.40	76.67	71.33	78.67	89.33	89.17	88.00	84.83	93.50	99.65	98.07	99.09	97.39	99.42	98.21	97.55
9-NN	90.83	87.67	84.83	86.50	87.70	76.83	71.17	79.00	89.33	89.17	87.50	84.67	92.83	99.65	98.07	99.32	97.19	99.42	98.39	97.10
11-NN	91.00	86.83	85.33	85.83	88.05	76.67	72.83	77.00	89.17	88.67	86.67	83.50	93.17	99.48	98.07	99.32	96.99	99.42	97.85	97.10
13-NN	91.00	87.17	84.50	85.33	87.87	75.33	73.50	77.17	88.33	89.33	87.50	83.50	93.50	99.48	98.07	99.09	97.39	99.22	97.67	97.10
SVM	90.50	89.67	90.33	87.00	87.35	76.50	79.00	80.50	90.00	88.50	88.83	93.17	92.00	98.96	99.42	98.41	96.99	99.81	99.46	97.77
LDA	89.50	88.50	89.50	87.17	87.52	76.00	79.33	81.67	89.67	87.83	88.83	90.00	92.50	99.31	98.84	99.09	96.79	99.61	99.10	97.77

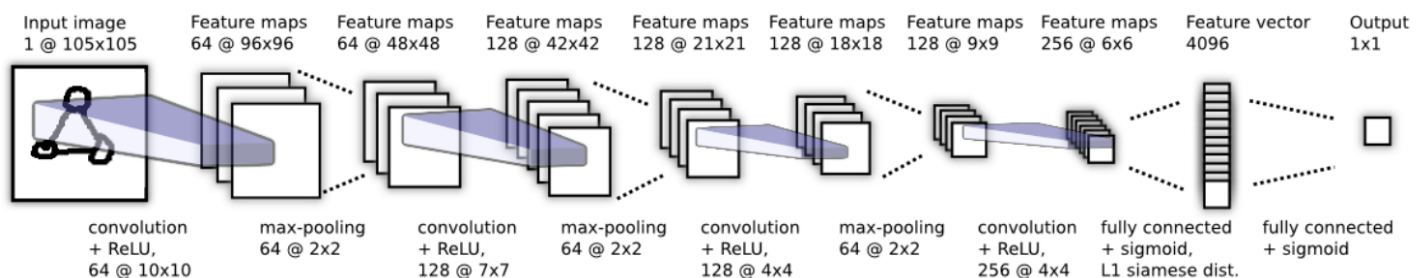
CELEBA Vs DeepNetworks				
	Kernel Size			
	3x3	4x4	5x5	7x7
3-NN	<b>89.96</b>	84.90	80.76	82.69
5-NN	<b>90.22</b>	86.63	82.48	82.77
7-NN	<b>89.57</b>	87.12	82.48	84.27
9-NN	<b>89.51</b>	86.73	83.31	84.27
11-NN	<b>89.25</b>	87.21	83.69	83.97
13-NN	<b>89.57</b>	87.31	84.20	83.45
SVMLinear	88.02	<b>88.75</b>	86.05	85.85
SVMsigmoid	<b>86.08</b>	72.60	83.38	63.66
SVMrbf	<b>89.77</b>	89.71	86.24	87.43
SVMPoly	82.51	86.06	84.65	<b>86.61</b>
LDA	87.56	<b>88.65</b>	86.11	85.48

Figure 4: CELEBA vs. Deep Networks

## 2.2 Deepfake Image Detection [2]

Deepfake detection can be a binary classification task, in this paper author used 2-phase learning architecture for deepfake detection and author used Siamese Neural Networks with CNN for the classification.[4]

### Siamese Neural Networks for One-shot Image Recognition



The dataset used in this article is DFDC (Deepfake Detection Challenge) consisting of 470GB of mp4 videos, 83% of the videos are deepfakes.

The author has used ResNet-18 CNN model trained as Siamese Network. The second name of this network is Common Fake Feature Network as shown in Figure 5.

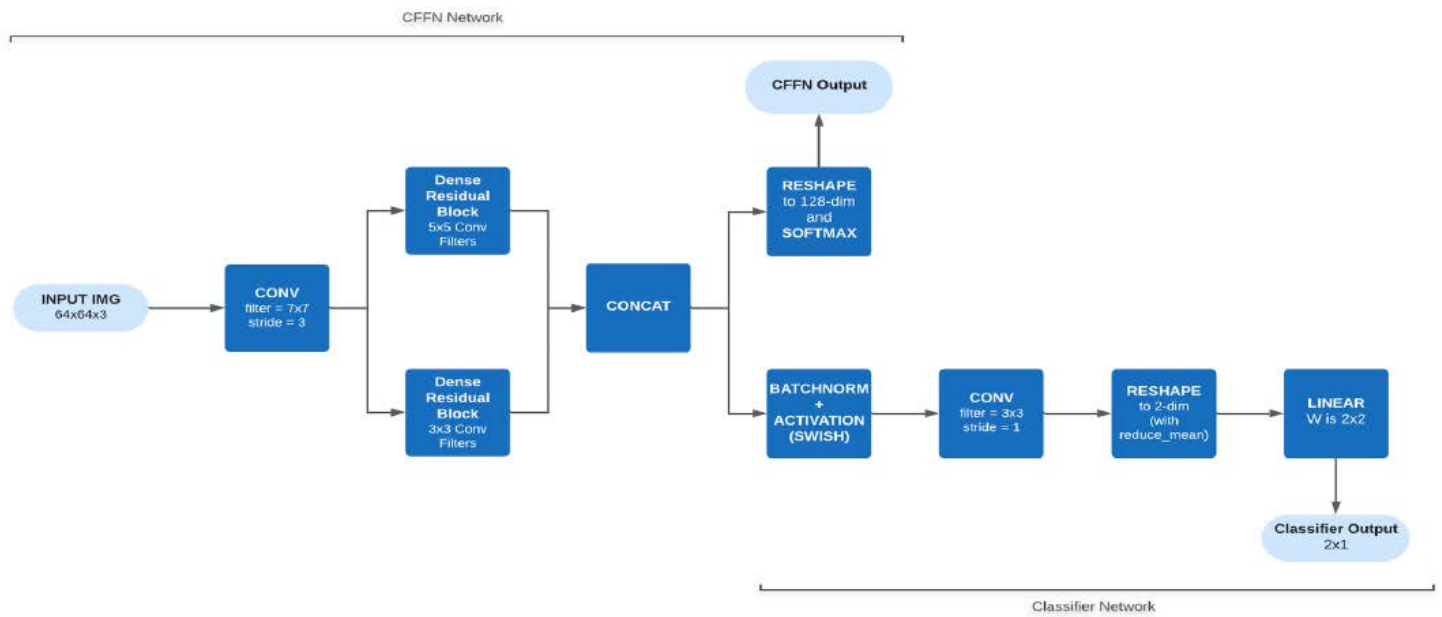


Figure 5: Proposed Method

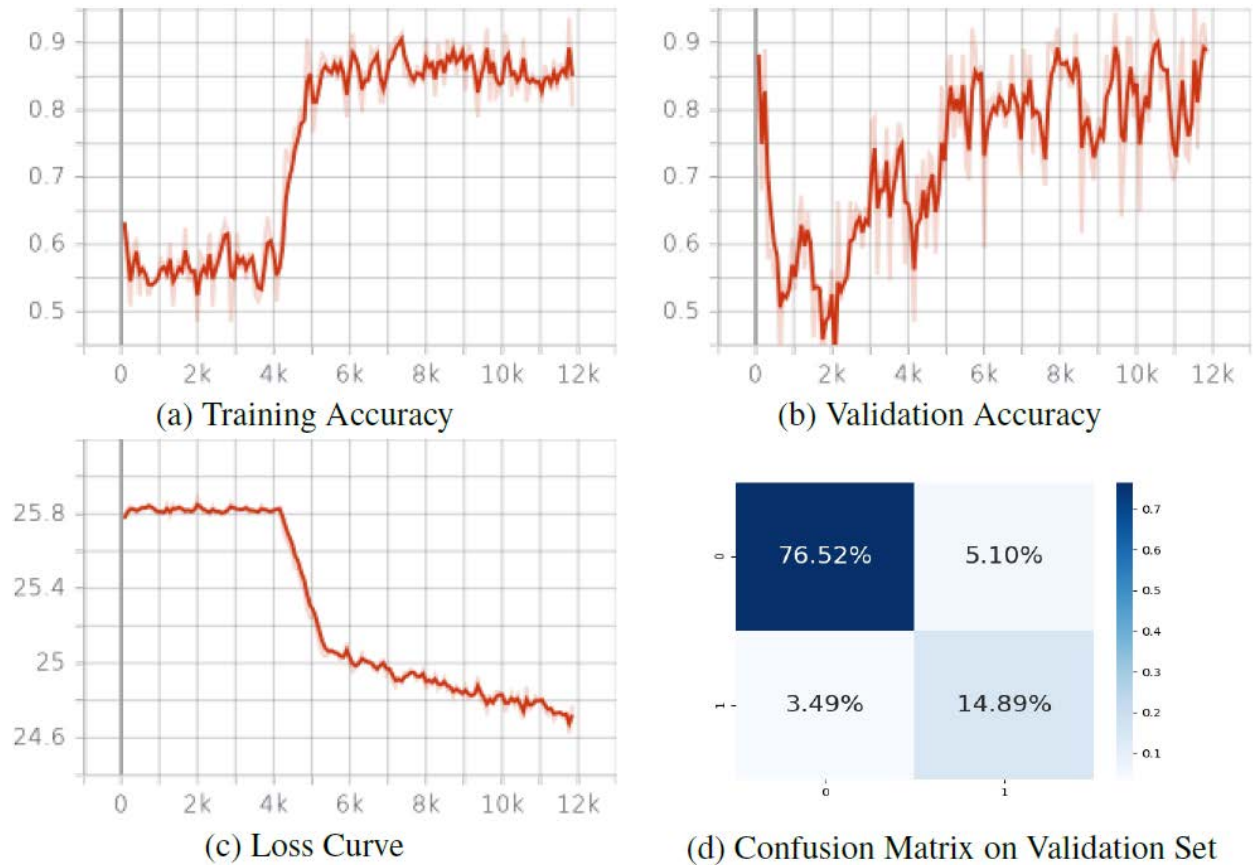
This model is trained for triple loss to learn feature- level distinctions between fake and real image.

$$L(A, P, N) = \max(0, ||f(A) - f(P)||_2^2 - ||f(A) - f(N)||_2^2 + \alpha)$$

Author has used a small CNN after the CFFN to binary classification trained for cross-entropy for high performance classifier.

The results obtained by the author are shown below:

Validation Accuracy	Training Accuracy	Precision	Recall	F1 Score
0.9141	0.9375	0.937516	0.956400	0.946864



### 2.3 Deepfakes Creation and Detection Using Deep Learning [5].

Detecting and classifying Deepfakes are crucial as it imposes threat to the future in form of fake news, imagine seeing a public figure making statement on record but you are not aware whether the video is fake or real.

The author explores Mesonet CNN based architecture for deepfake detection. Mesonet is designed specifically to detect deepfakes. While commonly deepfake videos are shared by online sources like social media applications the quality of deepfakes is commonly compressed form of the video, MesoNet is immune to this problem and can extract high level features that are not visible to the human eyes.

Autor use various image enhancement techniques to increase video quality before feeding it to their model to classify a video between REAL/FAKE.

After training the model CNN was able to detect deepfake with over 80% of the confidence as shown in Figure 6, but it faces challenges where the image is a face at an angle as shown in Figure 7.



Model confidence:  
0.7523



Model confidence:  
0.9979



Model confidence:  
0.7998



Model confidence:  
0.5786



Model confidence:  
0.9470



Model confidence:  
0.9601

*Figure 6: Results*



Model confidence:  
0.9859



Model confidence:  
0.9200



Model confidence:  
0.9997



Model confidence:  
0.9802



Model confidence:  
0.8301



Model confidence:  
0.9448

*Figure 7: Results with face at angle*

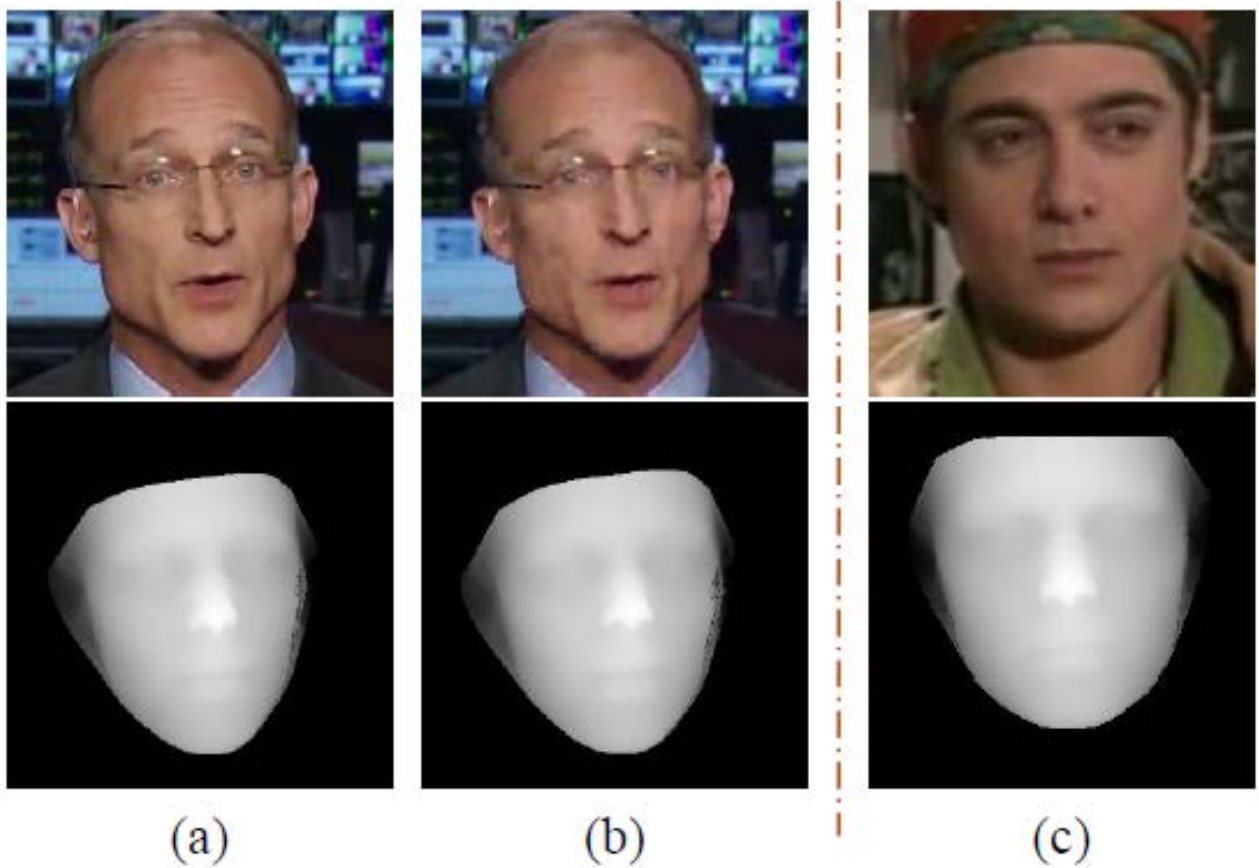


## 2.4 Exploring Depth Information for Face Manipulation Detection <sup>[3]</sup>

Deepfake detection is not only limited to the binary classification methods, but it can also use more advanced methods to explore deepfakes like using unique characteristics of an individual's face like depth of a certain area, face vectors (distance between two face points) etc.

Here author used face depth map for an individual to compare between it is real and deep fake images as

1. Face depth maps are immune to compression techniques.
2. Face depth maps are unique for a given face of individual.



*Figure 8: Face Depth Maps*

As shown in Figure 8 the face depth maps can be extracted from an individual's face, author has utilized the architecture given in Figure 9

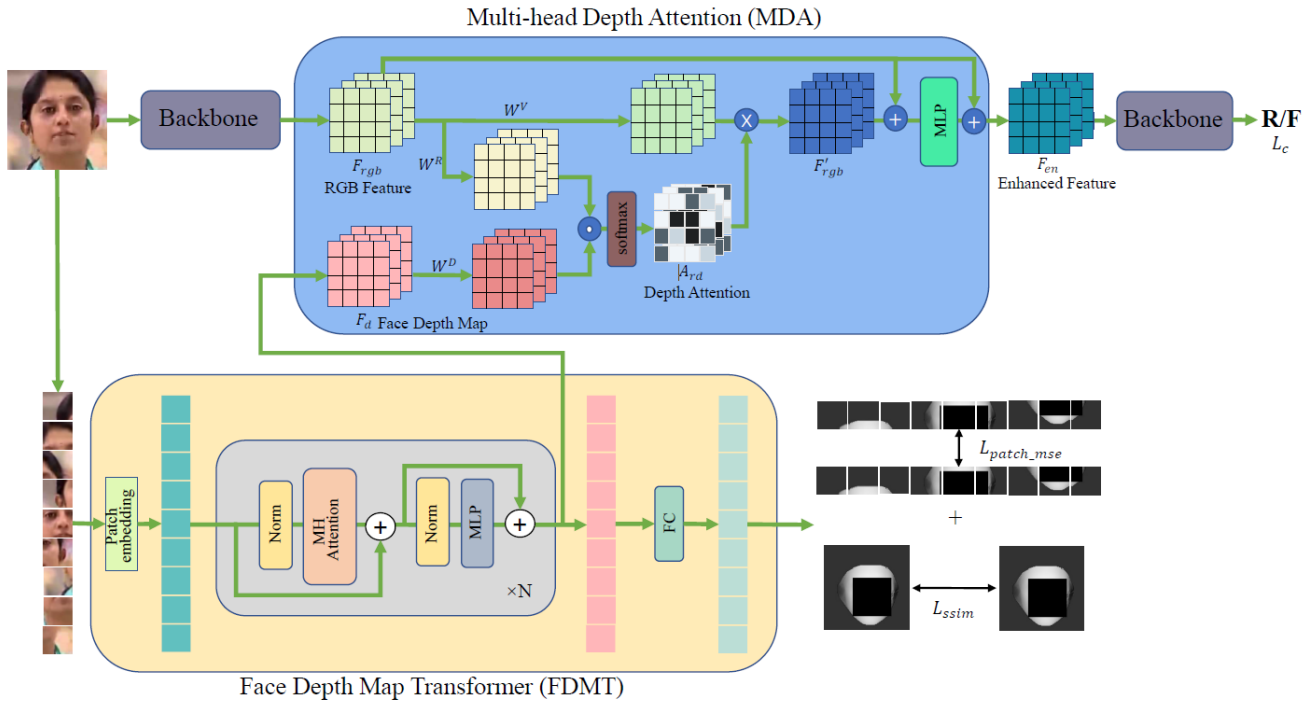


Figure 9: Proposed method for face manipulation detection

It was observed that in manipulating of image, face depth maps are altered, author has taken benefit from it to detect whether the give image is real or fake for given individual, the comparison of face depth map for real and fake image is shown in Figure 10



Figure 10: Face maps for REAL and FAKE image

The author's proposed model was able to perform 6.6% higher than the existing schemas.

## **CHAPTER 3**

### **MODELS AND ALGORITHMS**

This phase of project focused on the CNN based approach to classify between REAL/FAKE image, in other words this phase of project is binary classification task. The following sections explain the steps taken to build the binary classification task.

#### **3.1 DATASET**

The dataset used in this project is CELEB-DF V1 & V2, available publicly. The dataset contains the following folder architecture:

1. Celeb -real: contains 158 mp4 files obtained from YouTube.
2. Celeb – synthesized: contains 795 mp4 files obtained after deepfake generation.

The dataset is prepared for seventeen actors and has an average length of 16 seconds.

#### **3.2 PRE-PROCESSING**

The following steps are taken for preprocessing the dataset:

1. Frames from the videos are extracted at a specific frame rate.
2. The faces of actors are detected in the frames.
3. Frames with no faces are removed.
4. Frames with faces are cropped and saved as size (100,100).
5. Real and Fake frames are labeled as 0 -> REAL and 1 -> FAKE.

#### **3.3 MODEL**

1. Dataset has been split into 60:20:20 for Train, Test and Validation set, respectively.
2. A CNN and LSTM based model was created for binary classification of REAL/FAKE images.
3. The CNN model has 2 Convolution layers with kernel size 3 X 3 and MaxPooling layer after every convolution layer.
4. The output of CNN then fed into LSTM attention layer with sixty-four nodes.
5. Adam's optimizer is used in the model.

The architecture of the created model is shown in Figure 11 below.

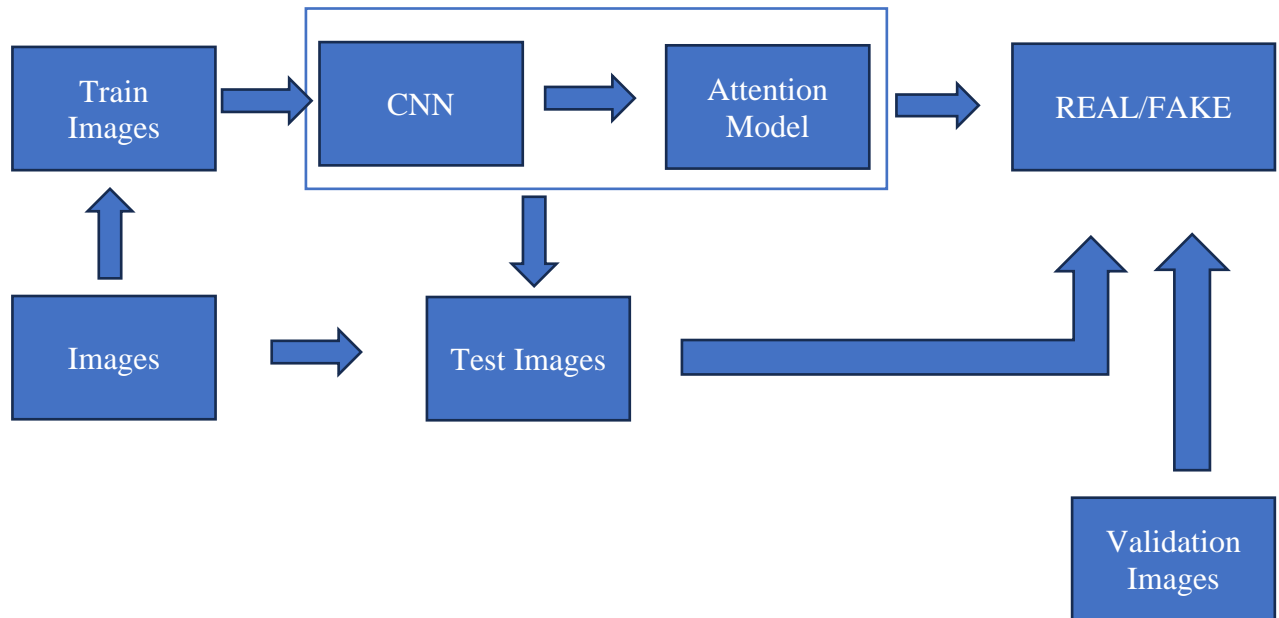


Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 100, 100, 3)]	0
conv2d (Conv2D)	(None, 98, 98, 32)	896
max_pooling2d (MaxPooling2D)	(None, 49, 49, 32)	0
conv2d_1 (Conv2D)	(None, 47, 47, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 23, 23, 64)	0
flatten (Flatten)	(None, 33856)	0
dense (Dense)	(None, 128)	4333696
reshape (Reshape)	(None, 1, 128)	0
bidirectional (Bidirectional)	(None, 1, 128)	98816
attention (Attention)	(None, 1, 128)	0
bidirectional_1 (Bidirectional)	(None, 128)	98816
dense_1 (Dense)	(None, 1)	129
=====		
Total params: 4,550,849		
Trainable params: 4,550,849		
Non-trainable params: 0		

Figure 11: Created CNN + LSTM Model

### 3.4 EXPERIMENTS

The Hyperparameters like learning rate, alpha value for the model and optimizers are changed and results are taken not consideration, data split values are also changed and compressed images are also taken in consideration.



## CHAPTER 4

### RESULTS AND FUTURE SCOPE

For the Evaluation and results for proposed model, accuracy and confusion matrices are taken into account, the results are compared for learning rate 0.001, 0.005 and 0.007 respectively for Adam's optimizer, the learning rate value define how fast a model can learn, but too high learning rate leads to unstable jumps over local minima / maxima of the loss function and too slow learning rate leads to slow learning, an optimal learning rate can move along with the local optima of the function and can extract local features also, these features are crucial for classification of the face images.

The following table shows the obtained results:

Learning Rate	Test Accuracy	Validation Accuracy	Confusion Matrix
0.001	98.51%	0.9819	$\begin{bmatrix} 1444 & 9 \\ 29 & 1063 \end{bmatrix}$
0.005	98.07%	0.9753	$\begin{bmatrix} 1446 & 7 \\ 50 & 1042 \end{bmatrix}$
0.007	96.54%	0.9603	$\begin{bmatrix} 1393 & 60 \\ 18 & 1074 \end{bmatrix}$

The Training and Validation loss curves are shown below:

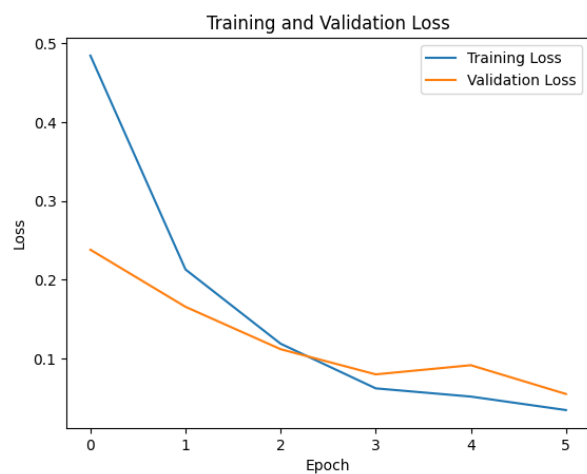


Figure 12: For LR 0.001

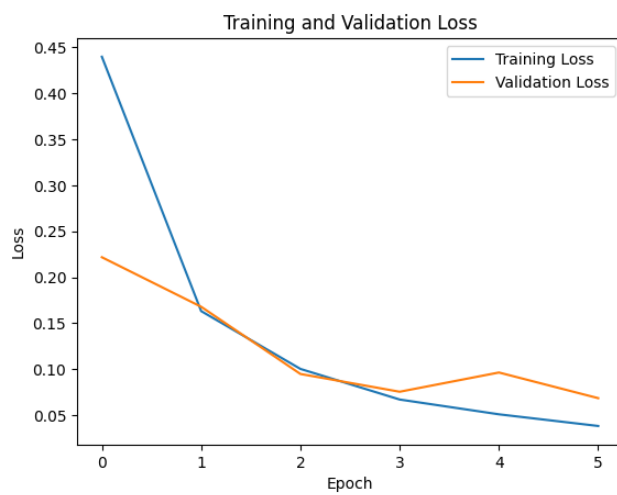


Figure 13: For LR 0.005

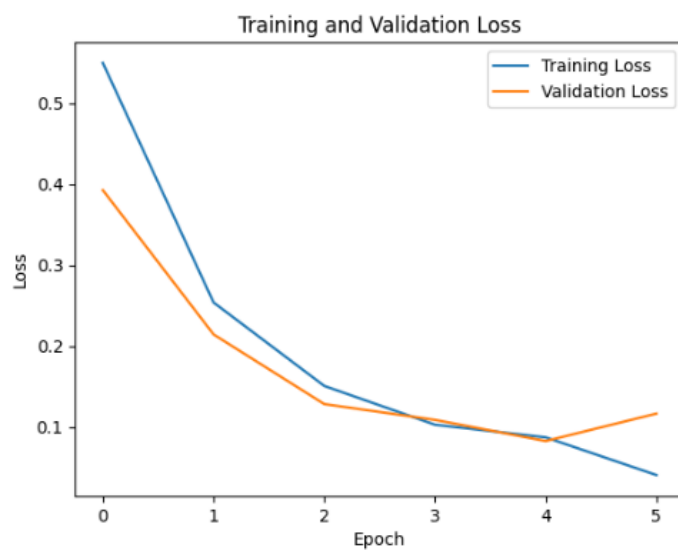


Figure 14: For LR 0.007

The Future scope of this project explore more detection techniques for deepfake detection, some of them are:

- Use of pre-trained deep learning models.
- Use of feature-based detection (face depth, face vectors etc.)
- Experiments and comparison of the developed models like compressed images, image enhancements, hyperparameter tuning etc.

## **CHAPTER 5**

### **CONCLUSION**

To summarize, the field of deepfake detection is characterized by a dynamic interaction between the constant development of synthetic media technology and the never-ending search for ways to ensure the authenticity of digital material. Deepfakes are ubiquitous, and we need to strengthen our defenses against their potentially harmful applications, as our investigation into them has shown.

The difficulty of detection is complex, as shown by the variety of deepfake kinds and the advanced methods used in their production. The armory against deepfakes is constantly growing, ranging from conventional techniques based on image and video analysis to state-of-the-art deep learning methods utilizing neural networks. But the path is not without difficulties: obtaining huge and varied datasets is necessary, and synthetic data is always changing.

## REFERENCES

- [1] Deepfake Detection by Analyzing Convolutional Traces (Luca Guarnera et al.)  
<https://arxiv.org/abs/2004.10448>
- [2] CS230: Deepfake Image Detection (Omkar Salpekar)  
[https://cs230.stanford.edu/projects\\_spring\\_2020/reports/38857501.pdf](https://cs230.stanford.edu/projects_spring_2020/reports/38857501.pdf)
- [3] Exploring Depth Information for Face Manipulation Detection (Haoyue Wang et al.)  
<https://arxiv.org/abs/2212.14230>
- [4] G. R. Koch, "Siamese neural networks for one-shot image recognition," 2015.  
<https://www.cs.cmu.edu/~rsalakhu/papers/oneshot1.pdf>
- [5] H. A. Khalil and S. A. Maged, "Deepfakes Creation and Detection Using Deep Learning," *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, Cairo, Egypt, 2021, pp. 1-4, Doi: 10.1109/MIUCC52538.2021.9447642.