Jyothi Hills, Panjal Road,
Vettikattiri PO, Cheruthuruthy , Thrissur,
Kerala 679531

Jyothi Engineering College
NAAC Accredited College with NBA Accredited Programmes*
Approved by AICTE & affiliated to APJ Abdul Kalam Technological University
A CENTRE OF EXCELLENCE IN SCIENCE & TECHNOLOGY BY THE CATHOLIC ARCHDIOCESE OF TRICHUR
JYOTHI HILLS, VETTIKATTIRI P.O, CHERUTHURUTHY, THRISSUR. PIN-679531 PH : +91- 4884-259000, 274423 FAX : 04884-274777
NBA accredited B.Tech Programmes in Computer Science & Engineering, Electronics & Communication Engineering, Electrical & Electronics Engineering and Mechanical Engineering valid for the academic years 2016-2022. NBA accredited B.Tech Programme in Civil Engineering valid for the academic years 2019-2022.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SEMINAR REPORT

Classification of Real and Fake Images

Submitted by

SANJANA S
JEC17CS088

Supervised by

Ms. ASWATHY WILSON
Asst. Prof., Dept. of CSE

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY (B.Tech)

in

COMPUTER SCIENCE & ENGINEERING
of

A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY

CREATING TECHNOLOGY LEADERS OF TOMORROW

DECEMBER 2020

Jyothi Hills, Panjal Road,
Vettikattiri PO, Cheruthuruthy , Thrissur,
Kerala 679531

Jyothi Engineering College
NAAC Accredited College with NBA Accredited Programmes*
Approved by AICTE & affiliated to APJ Abdul Kalam Technological University
A CENTRE OF EXCELLENCE IN SCIENCE & TECHNOLOGY BY THE CATHOLIC ARCHDIOCESE OF TRICHUR
JYOTHI HILLS, VETTIKATTIRI P.O, CHERUTHURUTHY, THRISSUR. PIN-679531 PH : +91- 4884-259000, 274423 FAX : 04884-274777
NBA accredited B.Tech Programmes in Computer Science & Engineering, Electronics & Communication Engineering, Electrical & Electronics Engineering and Mechanical Engineering valid for the academic years 2016-2022. NBA accredited B.Tech Programme in Civil Engineering valid for the academic years 2019-2022.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SEMINAR REPORT

Classification of Real and Fake Images

Submitted by

SANJANA S
JEC17CS088

Supervised by

Ms. ASWATHY WILSON
Asst. Prof., Dept. of CSE

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY (B.Tech)

in

COMPUTER SCIENCE & ENGINEERING
of

A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY

CREATING TECHNOLOGY LEADERS OF TOMORROW

DECEMBER 2020

Department of Computer Science and Engineering

JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY

THRISSUR 679 531

DECEMBER 2020

BONAFIDE CERTIFICATE

This is to certify that the seminar report entitled **Classification of Real and Fake Images** submitted by **Sanjana S (JEC17CS088)** in partial fulfilment of the requirements for the award of **Bachelor of Technology** degree in **Computer Science and Engineering** of **A P J Abdul Kalam Technological University** is the bonafide work carried out by her under our supervision and guidance.

Ms. Aswathy Wilson	**Dr. Swapna B Sasi**	**Dr. Vinith R**
Seminar Guide	Seminar Coordinator	Head of The Dept
Assistant Professor	Associate Professor	Professor
Dept. of CSE	Dept. of CSE	Dept. of CSE

DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

COLLEGE VISION

Creating eminent and ethical leaders through quality professional education with emphasis on holistic excellence.

COLLEGE MISSION

- To emerge as an institution par excellence of global standards by imparting quality engineering and other professional programmes with state-of- the-art facilities.
- To equip the students with appropriate skills for a meaningful career in the global scenario.
- To inculcate ethical values among students and ignite their passion for holistic excellence through social initiatives.
- To participate in the development of society through technology incubation, entrepreneurship and industry interaction.

DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

DEPARTMENT VISION

Creating eminent and ethical leaders in the domain of computational sciences through quality professional education with a focus on holistic learning and excellence.

DEPARTMENT MISSION

- To create technically competent and ethically conscious graduates in the field of Computer Science & Engineering by encouraging holistic learning and excellence.
- To prepare students for careers in Industry, Academia and the Government.
- To instill Entrepreneurial Orientation and research motivation among the students of the department.
- To emerge as a leader in education in the region by encouraging teaching, learning, industry and societal connect

PROGRAMME OUTCOMES (POs)

1. **Engineering Knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem Analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/Development of Solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct Investigations of Complex Problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern Tool Usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The Engineer and Society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and Sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and Team Work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project Management and Finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-Long Learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

1. The graduates shall have sound knowledge of Mathematics, Science, Engineering and Management to be able to offer practical software and hardware solutions for the problems of industry and society at large.
2. The grad uates shall be able to establish themselves as practising professionals, researchers or Entrepreneurs in computer science or allied areas and shall also be able to pursue higher education in reputed institutes.
3. The graduates shall be able to communicate effectively and work in multidisciplinary teams with team spirit demonstrating value driven and ethical leadership.

Programme Specific Outcomes (PSOs)

1. An ability to apply knowledge of data structures and algorithms appropriate to computational problems.
2. An ability to apply knowledge of operating systems, programming languages, data management, or networking principles to computational assignments.
3. An ability to apply design, development, maintenance or evaluation of software engineering principles in the construction of computer and software systems of varying complexity and quality.
4. An ability to understand concepts involved in modeling and design of computer science applications in a way that demonstrates comprehension of the fundamentals and tradeoffs involved in design choices.

Course Outcomes (COs)

C418.1 **Presentation Skills in terms of Content** : Students will be able to show competence in identifying relevant information, defining and explaining topics under discussion. They will demonstrate depth of understanding, use primary and secondary sources; they will demonstrate the working, complexity, insight, cogency, independent thought, relevance, and persuasiveness. They will be able to evaluate information and use and apply relevant theories.

C418.2 **Presentation Skills in terms of Organization** : Students will be able to show competence in working with a methodology, structuring their oral work, and synthesizing information. They will make a detailed study on the previous works related to their topic and will present the observations.

C418.3 **Presentation Skills in terms of Delivery** : Students will use appropriate registers and vocabulary, and will demonstrate command of voice modulation, voice projection, and pacing. They will be able to make use of visual, audio and audio-visual material to support their presentation, and will be able to speak cogently with or without notes.

C418.4 **Discussion Skills** : Students will be able to judge when to speak and how much to say, speak clearly and audibly in a manner appropriate to the subject, ask appropriate questions, use evidence to support claims, respond to a range of questions, take part in meaningful discussion to reach a shared understanding, speak with or without notes, show depth of understanding.

C418.5 **Listening Skills** : Students will demonstrate that they have paid close attention to what others say and can respond constructively. Through listening attentively, they will be able to build on discussion fruitfully, supporting and connecting with other discussants.

C418.6 **Argumentative Skills and Critical Thinking** : Students will develop persuasive speech, present information in a compelling, well-structured, and logical sequence, respond respectfully to opposing ideas, show depth of knowledge of complex subjects, and develop their ability to synthesize, evaluate and reflect on information.

Course Outcome							
Programme Outcomes		C418.1	C418.2	C418.3	C418.4	C418.5	C418.6
	1	3	3	3	3	3	3
	2	3	3	3	3	3	3
	3	3	3	3	3	3	3
	4	3	3	3	3	3	3
	5	3	3	3	3	3	3
	6	3	3	3	3	3	3
	7	3	3	3	3	3	3
	8	3	3	3	3	3	3
	9	3	3	3	3	3	3
	10	3	3	3	3	3	3
	11	3	3	3	3	3	3
	12	3	3	3	3	3	3

PO - CO Mapping

PEO - CO Mapping

Course Outcome							
Programme Educational Objective		C418.1	C418.2	C418.3	C418.4	C418.5	C418.6
	1	3	3	1	1	-	2
	2	3	3	3	3	1	3
	3	1	2	3	3	1	3

PSO - CO Mapping

Course Outcome							
Programme Specific Outcomes		C418.1	C418.2	C418.3	C418.4	C418.5	C418.6
	1	3	3	3	3	3	3
	2	3	3	3	3	3	3
	3	3	3	3	3	3	3
	4	3	3	3	3	3	3

Seminar Outcome

1. Studied about the concept of Deep Learning.
2. Studied about different Variational Encoders.
3. Analyzed and compared the general architecture of OC-VAE,OC FakeDect1 and OC FakeDect2.
4. Studied about different deepfake detection methods.
5. Analyzed the methodology of deepfake detection using One Class based Methods

Seminar Outcome - CO Mapping

	Course Outcome						
Seminar Outcome		C418.1	C418.2	C418.3	C418.4	C418.5	C418.6
	1	3	3	3	1	3	3
	2	3	3	1	1	3	3
	3	3	3	3	1	3	1
	4	3	3	3	3	1	1
	5	3	1	3	3	1	1

ACKNOWLEDGEMENT

I take this opportunity to express my heartfelt gratitude to all respected personalities who had guided, inspired and helped me in the successful completion of this seminar. First and foremost, I express my thanks to **The Lord Almighty** for guiding me in this endeavour and making it a success.

I take immense pleasure in thanking the **Management** of Jyothi Engineering College and **Dr. Sunny Joseph Kalayathankal**, Principal, Jyothi Engineering College for having permitted me to carry out this seminar. My sincere thanks to **Dr. Vinith R**, Head of the Department of Computer Science and Engineering for permitting me to make use of the facilities available in the department to carry out the seminar successfully.

I express my sincere gratitude to **Mr. Shaiju Paul & Dr. Swapna B Sasi**, Seminar Coordinators for their invaluable supervision and timely suggestions. I am very happy to express my deepest gratitude to my mentor **Ms. Aswathy Wilson**, Associate Professor,Department of Computer Science and Engineering, Jyothi Engineering College for his able guidance and continuous encouragement.

Last but not least, I extend my gratefulness to all teaching and non-teaching staff who directly or indirectly involved in the successful completion of this seminar work and to all friends who have patiently extended all sorts of help for accomplishing this undertaking.

ABSTRACT

Deepfake videos are AI generated videos that look real but are actually fake. The great threat of deepfake is that it is impossible for us to identify whether it is fake or not using our eyes as it is much perfect.Using such videos and images it is easy for malicious abuser to create arbitrary fake news and fool and mislead the public. Therefore, a new challenge of detecting Deepfakes arises to protect individuals from potential misuses. Many researchers have proposed various binary-classification based detection approaches to detect deepfakes. Generally Binary Classification methods are used for classification of real and fake images. But it requires large dataset of real and fake images for training the model. It is challenging to collect sufficient fake images in advance. When new deepfake generation methods are introduced, only little deepfake dataset are available for training the model. So the output of such models won't be accurate. This paper propose OC-FakeDect, which uses a one-class Variational Autoencoder (VAE) to train only on real face images and detects non-real images such as deepfakes by treating them as anomalies. This method achieved 97.5 percent accuracy in the NeuralTextures data of the well-known FaceForensics++ benchmark dataset without using any fake images for the training process.

Keywords - Detection, Identification, Convolutional Recurrent Neural Networks, Recurrent Neural Networks, Deep Learning,Variational Autoencoders

CONTENTS

List of Figures

List of Abbreviations

CNN	: *Convolutional Neural Network*
RNN	: *Recurrent Neural Network*
RMSE	: *ReconstructionScore*
MTCNN	: *Multi Task Cascaded Convolutional Neural Networks*
DF	: *Deep Fake*
AI	: *Artificial Intelligence*
GAN	: *Generative Adversarial Network*
VAE	: *Variational Autoencoder*
OC	: *One Class*
NT	: *Neural Textures*

CHAPTER 1
INTRODUCTION

1.1 Overview

Nowadays, people faced an emerging problem of AI synthesized face swapping videos, widely known as the DeepFakes. These kind of videos and images can be created to cause threats to privacy, fraudulence and so on. The idea of substituting face on photo is not so new as we can suggest. However, when the idea of neural networks became popular and humanity improved its computational skills, people began to use this technology in their everyday life.

1.1.1 Neural Network

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature. Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria.

1.1.2 Artificial intelligence

AI refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving. The ideal characteristic of artificial intelligence is its ability to rationalize and take actions that have the best chance of achieving a specific goal.

1.1.3 Deep Learning

Deep learning is an artificial intelligence (AI) function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural network.

1.1.4 CNN

:Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification. ConvNets have been successful in identifying faces, objects and traffic signs apart from powering vision in robots and self driving cars.Convolutional Neural Networks are very similar to ordinary Neural Networks. They are made up of neurons that have learnable weights and biases.

Nowadays we can download and run such programs - they can help us to get experience by experimenting without having a Ph.D. in math, computer theory, psychology, and more. When such technology became public domain, people immediately began to use it to create inappropriate content. For example, fake celebrity pornographic videos or revenge porn. But such videos and photos are more like baby pranks and no more. Great threat of Deepfake content came later. When it became hard to recognize by people eyes whether video or image was changed or not, real threats – fraudulent videos and images have come. This situation raises serious security and privacy concerns: imagine hackers that can use deepfakes to present a forged video of an eminent person to send out false and potentially dangerous messages to the public. Nowadays, fake news has become an issue as well, due to the spread of misleading information via traditional news media or online social media and Deepfake videos can be combined to create arbitrary fake news to support the agenda of a malicious abuser to fool or mislead the public. Therefore, a new challenge of detecting Deepfakes arises to protect individuals from potential misuses. Many researchers have proposed various binary-classification based detection approaches to detect deepfakes. But it requires large dataset of real and fake images for training the model. It is challenging to collect sufficient fake images in advance.

To overcome these data scarcity limitations, we formulate deepfakes detection as a one-class anomaly detection problem. This paper proposes OC-FakeDect, which uses a one-class Variational Autoencoder (VAE) to train only on real face images and detects non-real images such as deepfakes by treating them as anomalies.

1.2 Objective

The main objective of this seminar is to introduce a cost effective method that can overcome data scarcity limitation to detect and classify real and fake images so that it helps in preventing the usage of deepfakes in creating political distress,blackmailing, fake terrorism events, etc and helps to protect identity and privacy of a person.

1.3 Organization of The Report

The report is organised as follows:

- **Chapter 1: Introduction** Gives an introduction about Deepfakes and about advanced technologies that help to create and detect Deepfakes.
- **Chapter 2:Literature Survey**Summarizes different Deepfake detection Techniques.
- **Chapter 3: Classification of Real and Fake Images** Discusses in Depth about creation of Deepfakes, its disadvantages and its cost effective detection approach.
- **Chapter 4: Advantages and Limitations** Disscuss about advantages and limitations of One-Class approach.
- **Chapter 5: Conclusion** The overall detection method and inferred results are included.
- **References** Includes references for future purpose.

CHAPTER 2
LITERATURE SURVEY

2.1 Image Feature Detection for Deepfake Video Detection

Detecting DeepFake videos are one of the challenges in digital media forensics. This is a method to detect deepfake videos using Support Vector Machine (SVM) regression. The SVM classifier can be trained with feature points extracted using one of the different feature-point detectors such as HOG, ORB, BRISK, KAZE, SURF, and FAST algorithms. A comprehensive test of the proposed method is conducted using a dataset of original and fake videos from the literature. Different feature point detectors are tested. The result shows that the proposed method of using feature-detector-descriptors for training the SVM can be effectively used to detect false videos.[4]

Here, HOG, ORB, BRISK, KAZE, SURF, and FAST algorithms are tested and compared for detecting DeepFake videos. The performance of the selected feature detector descriptors are investigated using Support Vector Machine (SVM) regression .

- **Support Vector Machine:** Support vector machine (SVM) is one of the powerful statistical techniques used as a classification and regression tool. SVM was used in pattern recognition with the help of some image processing filters. Their enhanced method resulted in a promising effective tool to be used in removing the noise from Electrocardiographic signals.
- **HOG:** Histogram of Oriented Gradient (HOG) is an effective descriptor method that basically divides an image into blocks from which it uses the histogram gradient information to compute the edge direction.
- **ORB:** The main advantage of ORB is that it is rotation invariant and resistant to noise and small changes[9] .
- **BRISK:** This method is used to detect corners and edges. BRISK is also rotation invariant and resistant to noise and small changes[6].
- **KAZE:** The algorithm works on retaining the boundaries of an object in images and reduces the noise. Thus, it has “more distinctiveness at varying scales with the cost of moderate increase in computational time”.

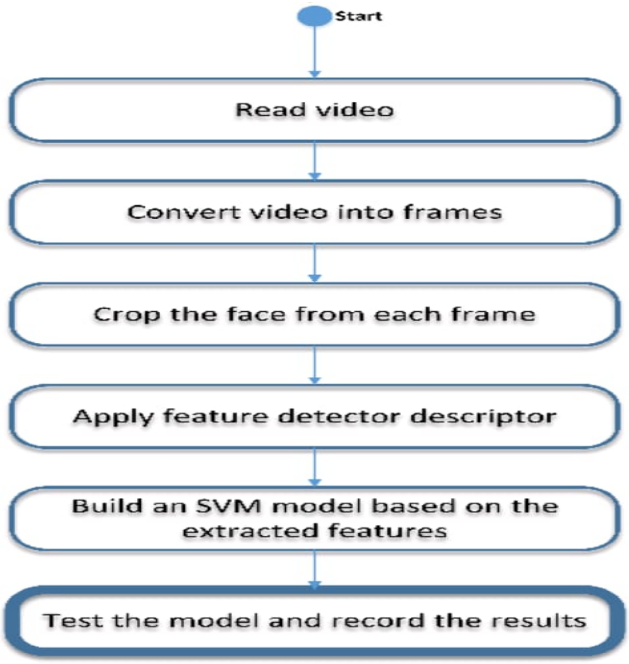


Figure 2.1: Detection Method

- **SURF:** Speeded-Up Robust Features (SURF) is a method that depends on Gaussian scale space analysis of an image, based on sums of Haar wavelet components, and uses integral images to enhance feature-detection speed .
- **FAST:** Features from Accelerated Segment Test (FAST) was proposed to solve the problem of complexity for real-time applications. It was proven that FAST is applicable with different views of a 3D scenes as well.

First, the input video is transformed into a sequence of frames. In order to speed up the process, only a few frames per second are extracted. In addition, for each extracted frame, the auto-face detection algorithm is used to identify and crop the face into a rectangular area. This normally reduces the image size heavily for faster processing. Then, the desired feature point detection method is used to extract point descriptors which will be aggregated across the different frames and fed into the SVM classification model for training or detection. Dataset used here is special dataset for deepfake videos and it contains 98 videos. Half of them are real and half of them are fake. 85 percent and 15 percent of the data set are used for training and testing respectively. For the training set, each video is converted into a set of frames where 5 frames are extracted per second. For each frame, a face detection algorithm is used to detect and crop the face into a 200×200 pixels sub-image. The different feature-detection algorithms are run on the extracted set of cropped faces to extract the feature points descriptors. The accumulative results for all the images will be used to build the SVM model. The testing set

is used to test the accuracy of the produced model and to record the result in terms of the confusion matrix. All videos are divided into frames. All parameters for BRISK, KAZE, FAST and SURF algorithms were set to default values. The cell size in the HOG algorithm was set to 4×4, and the scale factor and cell size in the ORB algorithm is set to 1.000001 and 100, respectively. Confusion matrix is used to compare results . Results are as follows. Feature detection algorithm of HOG provides the best performance with accuracy of 94.5 percentage. SURF and ORB also provides good accuracy exdeeding 90 percentage. KAZE, on the other hand was the least effective with an accuracy of 76.5 percentage. BRISK and FAST scores above 86.5 percent accuracy.

2.2 Detecting Deepfakes with Metric Learning

With the arrival of several face-swapping applica-tions such asFaceApp, SnapChat, MixBooth, Face Blender and many more, the authenticity of digital media content is hanging on a very loose thread. On social media platforms, videos arewidely circulated often at a high compression factor. In this work, analyzation of several deep learning approaches in the context of deepfakes classification in high compression scenarios are done and demonstrate that a proposed approach based on metric learning can be very effective in performing such a classification. Metric learning is an approach based directly on a distance metric that aims to establish similarity or dissimilarity between objects. While metric learning aims to reduce the distance between similar objects, it also aims to increase the distance between dissimilar objects. Metric learning can be very effective in classification of deep fakes.It learns to enhance the feature space distance between the cluster of real and fake videos embedding vectors. Multitask Cascaded CNNs (MTCNN) is used to extract faces out of frames MTCNN is a neural network which detects faces and facial landmarks on images.[5]Final architecture uses triplet network to discriminate between fake and real videos. MTCNN extracts faces from frames. Facenet generates 512 dimension embeddings for each face in the feature space. As facenet is developed for face recognition, each unique face occupies a small cluster in the feature space. Using Semi-Hard triplets, the embeddings of fake frames are distinctly separated through triplet loss.Extracted faces from video frames. Triplet network is used to discriminate between fake and real videos FaceNet is used to generate face embeddings. Using semi-hard triplet networks, embeddings of fake frames are distinctly separated through triplet loss. Fake videos and real videos are classified distinctly.

Figure 2.2: Extraction of face from frame using MTCNN algorithm.

2.3 Deepfake Video Detection Using Recurrent Neural Networks

In recent months a machine learning based free software tool has made it easy to create believable face swaps in videos that leaves few traces of manipulation, in what are known as "deepfake" videos. Scenarios where these realistic fake videos are used to create political distress, black-mail someone or fake terrorism events are easily envisioned.This paper proposes a temporal-aware pipeline to automatically detect deepfake videos. Our system uses a convolutional neural network (CNN) to extract frame-level features. These features are then used to train a recurrent neural net-work (RNN) that learns to classify if a video has been subject to manipulation or not.

Two sets of training images are required. The first set only has samples of the original face that will be replaced, which can be extracted from the target video that will be manipulated. This first set of images can be further extended with images from other sources for more realistic results. The second set of images contains the desired face that will be swapped in the target video. To ease the training process of the autoencoders, the easiest face swap would have both the original face and target face under similar viewing and illumination conditions. However, this is usually not the case. Multiple camera views, differences in lightning conditions or simply the use of different video codecs makes it difficult for autoencoders to produce realistic faces under all conditions. This usually leads to swapped faces that are visually inconsistent with the rest of the scene. This frame-level scene inconsistency will be the first feature that will be exploited with this approach. If two autoencoders are trained separately on different sets of faces, their latent spaces and representations will be different. This means that each decoder is only able to decode a single kind of latent representations which it has learnt during the training

phase. This can be overcome by forcing the two set of autoencoders to share the weights for the encoder networks, yet using two different decoders.[1]When the training process is complete,latent representation of a face generated from the original subject present in the video is passed to the decoder network trained on faces of the subject we want to insert in the video.The decoder will try to reconstruct a face from the new subject, from the information relative to the original subject face present in the video. This process is repeated for every frame in the video where we want to do a face swapping operation. This is usually a second source of scene inconsistency between the swapped face and the re-set of the scene. Because the encoder is not aware of the skin or other scene information it is very common to have boundary effects due to a seamed fusion between the new face and the rest of the frame.The proposed system is composed by a convolutional LSTM structure for processing frame sequences. There are two essential components in a convolutional LSTM:

- CNN for frame feature extraction.
- LSTM for temporal sequence analysis

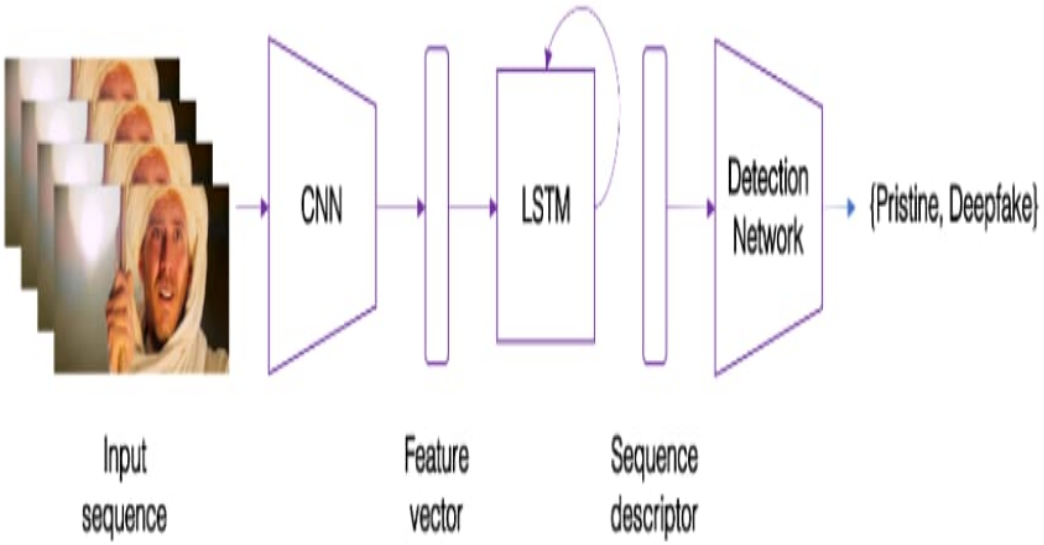


Figure 2.3: RNN Detection System

2.4 Effective and Fast DeepFake Detection Method Based on Haar Wavelet Transform

DeepFake using Generative Adversarial Networks (GANs) tampered videos reveals a new challenge in today's life. With the inception of GANs, generating high-quality fake videos becomes much easier and in a very realistic manner. Therefore, the development of efficient tools that can automatically detect these fake videos is of paramount importance DeepFake detection method takes the advantage of the fact that current DeepFake generation algorithms cannot generate face images with varied resolutions, it is only able to generate new faces with a limited size and resolution, a further distortion and blur is needed to match and fit the fake face with the background and surrounding context in the source video. This transformation causes exclusive blur inconsistency between the generated face and its background in the outcome DeepFake videos, in turn, these artifacts can be effectively spotted by examining the edge pixels in the wavelet domain of the faces in each frame compared to the rest of the frame. A blur inconsistency detection scheme relied on the type of edge and the analysis of its sharpness using Haar wavelet transform is used.Haar wavelet is a sequence of rescaled "square-shaped" functions which together form a wavelet family.[13]

Linear blurred images can be described as G = H * F + N (1). Where G, F N represent noisy image and Blur function is represented by H matrix. The blur extent is measured by testing the edge features in an image. The edge feature is one of these features that can be used. If the blur is present, both the edge sharpness and its type will be changed and that will indicate whether the face image has been manipulated or not.

Different types of edges are present in an image. Generally, there are three classes of edge type: Dirac-Structure, Step-Structure, and Roof-Structure.The Step-Structure type is divided according to the change of intensity whether it is gradual or not into: "A Step-Structure" and "G Step-Structure". Every image has all types of edges more or less, most of the G StepStructure and Roof-Structure are sharp enough. In case it is blurred, the edges lose their sharpness. The sharpness parameter is measured by the sharpness parameter alpha.This method detects whether a face image is blurred or not based on Dirac-structure and A Step-Structure.A blur extent is identified by taking sharpness of Roof-Structure and G Step-Structure into account. The sharpness of the edge is indicated by the parameter alpha , if alpha is larger, means the edge is sharper.

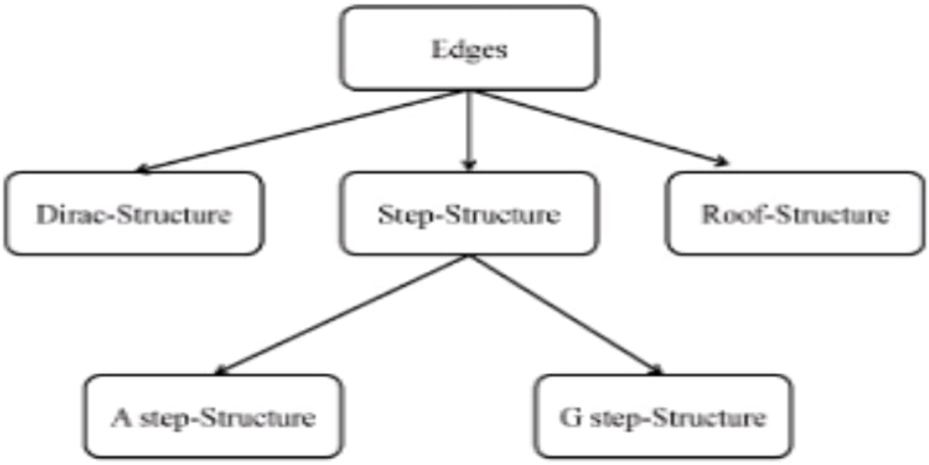


Figure 2.4: Edge Classification

The noise factor N in "(1)" can be neglected since there is a small noise ratio in photos usually acquired from digital cameras.The main blur functions H is a convolution operation that affects the equation and will change the edge property. Take into consideration that there will be no Dirac-Structure or A Step-Structure in the blurred image. On the other hand, both Roof-Structure and G Step-Structure will tend to lose their sharpness (less value).

Discrete Haar functions can be described as functions determined by sampling the Haar functions at 2n points. A matrix form can represent the function in a convenient means. Every row in the matrix H(n), have a discrete Haar sequence Haar(w, t) each alone, where the index (w) represents the number of the Haar function, the discrete point of the function determination interval is identified by index (t). By the following recurrence relation , any dimension of the Haar matrix can be gained;

$$H(n) = \begin{bmatrix} H(n-1) & \otimes [1 \quad 1] \\ 2^{\frac{n-1}{2}} I(n-1) & \otimes [1 \quad 1] \end{bmatrix}, H(0) = 1$$

Where $H(n)$ - the discrete Haar functions of degree 2^n matrix, $I(n)$ - identity matrix of degree 2^n.

Figure 2.5: Recurrence relation for Haar matrix

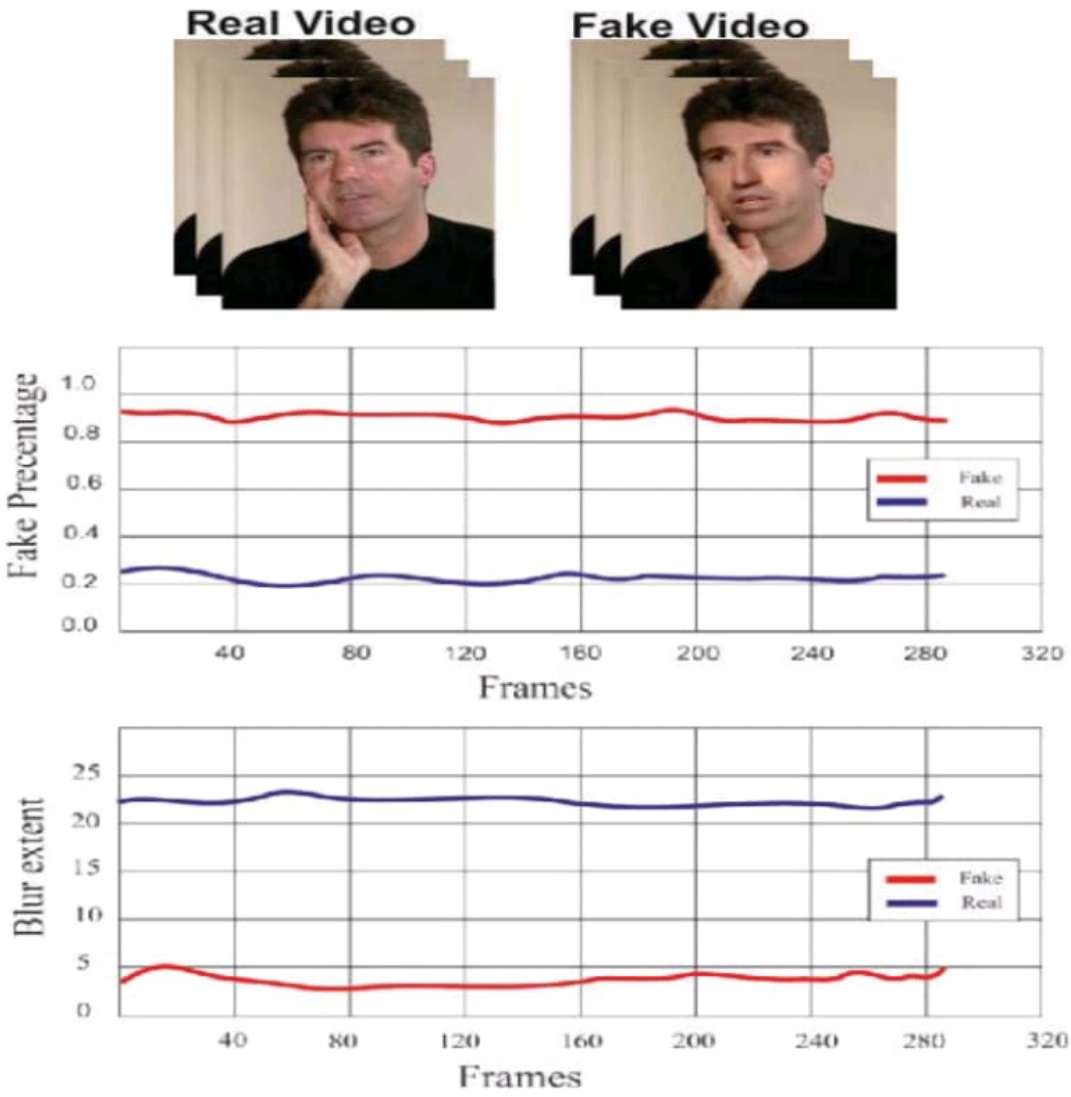


Figure 2.6: Example of the proposed method on one of the DeepFake generated videos from the "UADFV" datase

2.5 Adversarial Neural Network for Self-representation in One Class Classification

The one class classification problem is a special type of two-class classification. In the conventional classification problem, we distinguish the input data into one of several different categories, while in the OCC problem, we decide that the input data belongs to or does not belong to the target class. One of the most causes contributing for the development of the OCC is that some type of samples are cheaper to access while tagging on the negative is a manualintensive and time-consuming operation[12].Traditional OCC algorithms are based on hand-crafted features. However, it is hard to select discriminative features when the training data are all from the same category. With the rapid development of deep learning in pattern recognition, it makes the model to learn effective visual representations of training data itself rather than relying on the features selected manually. Here adversarial neural model called AVAE is proposed for OCC.

One class classification is closely related to many applications like outlier detection and anomaly detection. The main problem of OCC is that the extreme form of class imbalance where training instances are only available from the majority class and no information is available regarding the minority classes. Rather than learning to discriminate between classes, as in binary or multi-class classification, one class classifiers must learn to recognize the majority class with its unique feature. The best method to realize OCC by machine learning is to extract features and find a boundary of inter-class and outers. End-to-end AVAE network for one class classification is composed of two parts: reconstructor and discriminator. The reconstructor is used to learn the visual features of the target class images so as to reconstruct it and distort the non-target class images. The discriminator accomplishes one class classification by capturing the difference between the reconstructed target class images and the reconstructed non-target images. In addition, it also proposes an effective training method which includes two steps: joint training and vae-only training. The joint training is aim at training the discriminator to have certain ability to identify real and fake data, while the only training is to further improve the ability of reconstructor to reconstruct target class image very well. This also enables the discriminator to better classify target class and non-target class. The experimental results on CIFAR-10 shows that our method can efficiently perform one class classification on color images.[15]

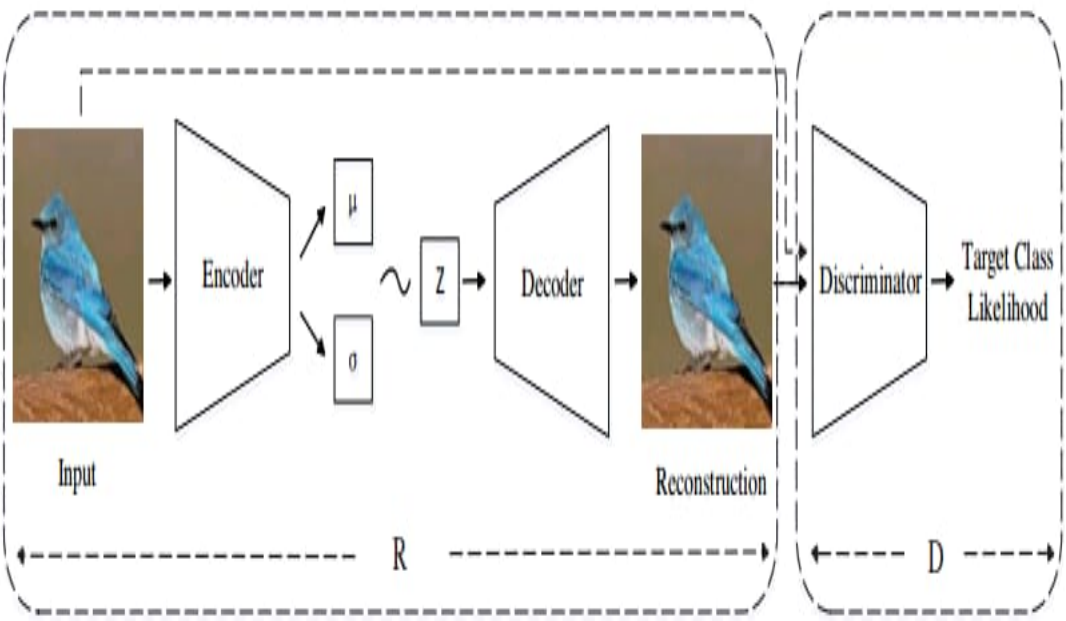


Figure 2.7: The framework of deep learning model for OCC

2.6 Digital Forensics and Analysis of Deepfake Videos

The spread of smartphones with high quality digital cameras in combination with easy access to a myriad of software apps for recording, editing and sharing videos and digital images in combination with deep learning AI platforms has spawned a new phenomenon of faking videos known as Deepfake.The novel method will be introduced to detect deepfake videos and to get high detection accuracy more effectively and efficiently than other, common methods. The DFT-MF model will be tested on two new datasets: The Deepfake Forensics (Celeb-DF) dataset and the Deepfake Vid-TIMIT dataset. The proposed method utilizes Convolutional Neural Networks (CNN) to export videos into frames (images) and subsequently convert these images into gray-scale images to be processed and classified within one of the various deep learning type. The CNN deep learning algorithm is used to classify deepfake videos. MoviePy, which is an open-source application is used for editing and cutting videos, to cut the video based on certain words in which the mouth appears open and the teeth are visible. We eliminated all images that are irrelevant to our thus saving time and resources. This is in contrast to other algorithms, that extract all images from the video and then attempt to identify the facial region within the extracted images. The DFT-MF model was built to detect deepfake videos by using the mouth as a biological signal. First, the datasets were used that contain both fake and real videos Celeb-DF and Deepfake-TIMIT. Secondly, deep learning (CNN) was applied to classify

fake videos, depending on the features that will be taken from the mouth as a biological signal.

In this stage, the information will be collected from relevant sources to create a new dataset that contains a combination of fake and real videos. The Deepfake Forensics (Celeb-DF) dataset and the Deepfake Vid-TIMIT dataset is used.Prior to performing analysis on the image frames, some preprocessing is required. Face detection is one of the most essential steps of this work to enable us to filter out image frames that do not contain faces. TO this end, the Dlib classifier will be used to detect face landmarks and eliminate all unnecessary frames.The DFT-MF model focuses on area surrounding the mouth, especially the teeth; therefore, the mouth area will be cropped from a face in the frame[2].

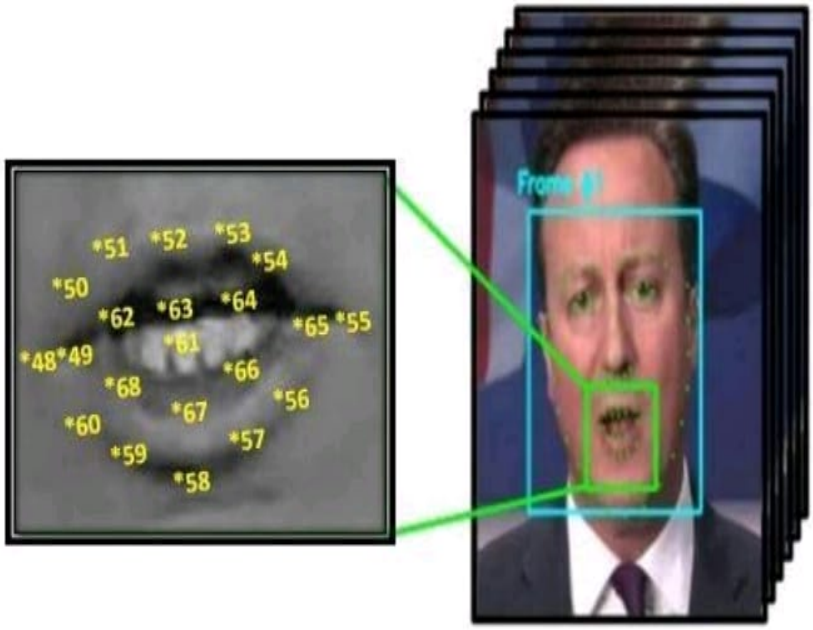


Figure 2.8: Mouth Cropping

This area will be used by DFT-MF model to crop the mouth region based on the ratio between each two-point upper lips and the lower lips. The next step is to exclude all frames that contain a closed mouth by calculating distances between lips. This is because an image with a closed mouth has no fake value as nothing is being uttered in that frame.The test data is split into fake and real videos for training 25000 Frames: 12500 frames were labeled as Real and 12500 frames were labeled as Fake videos.The DFT-MF model will use the deep learning supervised, Convolutional Neural Network (CNN), to classify videos into fake or real based on a (threshold) number of the fake frames that are identified in the entire video based on calculate three variables word per sentence, speech rate and frame rate.

2.7 Methods of Deepfake Detection Based on Machine Learning

Nowadays, people faced an emerging problem of AI synthesized face swapping videos, widely known as the DeepFakes. This kind of videos can be created to cause threats to privacy, fraudulence and so on. Sometimes good quality DeepFake videos recognition could be hard to distinguish with people eyes[7].

2.7.1 Usual Indicatiors of deepfake videos

1. Too smooth skin, lack of skin details – this indicators are consequence of one problem in DeepFake algorithms: low resolution of synthesized faces. But sometimes detection can be very hard, especially because of makeup on one of two faces.

2. Color mismatch between the synthesized face and the original face - this indicator can be used in human DeepFake recognition, but sometimes such mismatches can be very tricky to detect by eyes. But not for good program.

3. Visible parts of original face or temporal flickering - when face swapping algorithm got improper choice of the face region we can see artifacts of the original face or even whole original face flickering. May be it is just one frame of the whole one-hour video. But we should check this frame more precisely.

4. Head position – this indicator can appear due to the problem, described above.

5. Artifacts on small moving parts – due to resolution limits, DeepFake algorithm cannot produce small moving parts with good quality. That's why we can sometimes see artifacts on hairs, eyebrows, eyelashes or some small skin defects.

6. Eye blinking rate – indicator that was very useful in the very beginning of the face swapping algorithms popularity. Due to small datasets of photos and very small amount of eye-closed pictures there DeepFake couldn't produce an eye-blinking face and so blinking rate reduces. Now new versions of algorithms solved such problem, so it's not very helpful anymore.

7. Face warping artifacts indicator may be the best choice right now, but when new face swapping algorithm and technologies appear and higher quality face pictures will be synthesized it can become useless.

8. Person's patterns of behavior – can be useful, when we talk about the puppet-master and lip-sync techniques of Deepfakes. But it's very hard to use such indicator on photos and

it can detect only fakes with person whose behavior patterns were taken.

DenseNet169 with face warping artifacts indicator is used here. For extracting faces from the picture dlib package is used. After that random affine transformations are used randomly on resized pictures. Then random specific bluris added. Finally, face pictures are resized back and the picture is made whole again. Celeb-DF dataset is used to evaluate the model. It is one of the newest datasets of Deepfake videos. It has about 1 thousand videos with HQ face swapping algorithms used to synthesize part of them. There are good quality synthesized videos with almost none artifacts of original face, small moving parts and other indicators. So it can be really hard challenge for the model.

Model	Celeb-DF AUC
Two streams	55.7
HeadPose	54.8
FWA (ResNet50)	53.8
FWA (DenseNet169)	**60.1**

Figure 2.9: AUC Performance Percentage on CELEB-DF Dataset

From the table it is clear that DenseNet 169 outperform other methods.

CHAPTER 3
CLASSIFICATION OF REAL AND FAKE IMAGES

Deepfake videos are AI generated videos that look real but are actually fake. The great threat of deepfake is that it is impossible for us to identify whether it is fake or not using our eyes as it is much perfect.Using such videos and images it is easy for malicious abuser to create arbitrary fake news and fool and mislead the public.There is also positive use of deepfakes such as creating voices of those who have lost theirs or updating episodes of movies without reshooting them. However, the number of malicious uses of deepfakes largely dominates that of the positive ones.Less and less effort is required to produce a stunningly convincing tempered footage. Recent advances can even create a deepfake with just a still image. Generally Binary Classification methods are used for classification of real and fake images. But it requires large dataset of real and fake images for training the model. It is challenging to collect sufficient fake images in advance. When new deepfake generation methods are introduced, only little deepfake dataset are available for training the model. So the output of such models won't be accurate. This method requires only real images for training so that data scarcity limitation can be solved and gives output with 97.5 percent accuracy.

Figure 3.1: Example of real and deepfake image

CREATION OF DEEPFAKES:

The main ingredient in deepfakes is machine learning. Using machine learning it became possible to produce deepfakes much faster at a lower cost. To make a deepfake video of someone, a creator would first train a neural network on many hours of real video footage of the person to give it a realistic "understanding" of what he or she looks like from many angles and under different lighting. Then they'd combine the trained network with computer-graphics techniques to superimpose a copy of the person onto a different actor. While the addition of AI makes the process faster than it ever would have been before, it still takes time for this process to yield a believable composite that places a person into an entirely fictional situation. The creator must also manually tweak many of the trained program's parameters to avoid telltale blips and artifacts in the image. The process is hardly straightforward.[8]

Many people assume that a class of deep-learning algorithms called generative adversarial networks (GANs) will be the main engine of deepfakes development in the future. Deepfakes are created by superimposing an existing source image onto a target image using Autoencoders or GANS(Generative Adversarial Networks) to create new forged image. GAN-generated faces are near-impossible to tell from real faces. GANs are hard to work with and require a huge amount of training data. It takes the models longer to generate the images than it would with other techniques and most important GAN models are good for synthesizing images, but not for making videos. They have a hard time preserving temporal consistency, or keeping the same image aligned from one frame to the next. Methods like Adobe Photoshop, StyleGAN, Faceswap, PGGAN and diverse high-fidelity images with VQ-VAE-2 can also be used to create fake images. Current facial manipulation methods can be broadly categorized into the following categories:

1. Facial Expression manipulation
2. Identity manipulation

In facial expression manipulation one can transfer facial expressions of a person to another using a method such as Face2face and identity manipulation is based on face swapping methods, in which one can replace a person's face with that of another person.

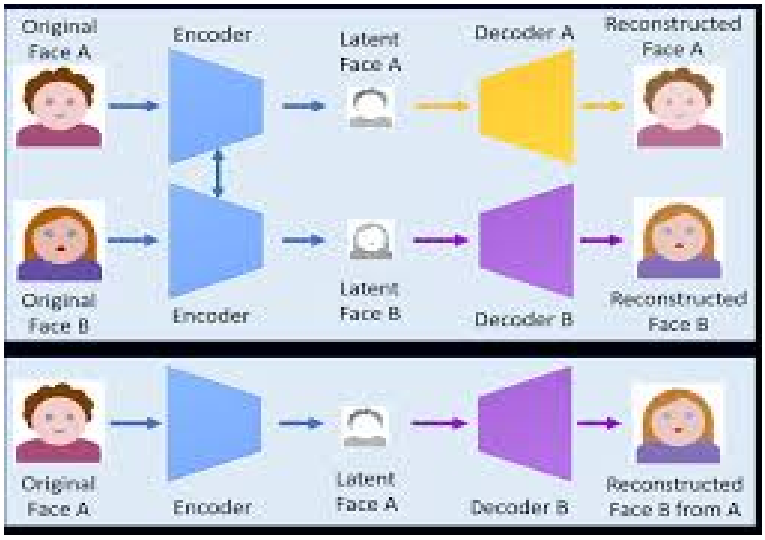


Figure 3.2: Creation of Deepfake image

DEEPFAKE DETECTION APPROACHES:

Deepfake detection is normally deemed a binary classification problem where classifiers are used to classify between authentic videos and tampered ones. This kind of methods requires a large database of real and fake videos to train classification models. The number of fake videos is increasingly available, but it is still limited in terms of setting a benchmark for validating various detection methods. To address this issue, Korshunov and Marcel produced a notable deepfake data set consisting of 620 videos based on the GAN model using the open source code Faceswap-GAN. Videos from the publicly available VidTIMIT database were used to generate low and high quality deepfake videos, which can effectively mimic the facial expressions, mouth movements, and eye blinking. These videos were then used to test various deepfake detection methods. Test results show that the popular face recognition systems based on VGG and Facenet, are unable to detect deepfakes effectively. Other methods such as lip-syncing approaches and image quality metrics with support vector machine (SVM) produce very high error rate when applied to detect deepfake videos from this newly produced data set. This raises concerns about the critical need of future development of more robust methods that can detect deepfakes from genuine.

Zhou et al proposed detection of face swapping manipulations of two types using a two-stream network. Raghavendra et al proposed a method to detect altered faces using two pre-trained deep Convolutional Neural Networks. They both require real and fake face image data to train their models. By increasing the layer depth of VGG16 and VGG19, these models can also be used for classification, but are very costly in terms of resource consumption, and are more difficult and time-consuming to train. ShallowNet and XceptionNet algorithms, which

can also classify real and fake images, showed promising results. ForensicTransfer addresses this issue using a smaller amount fake images and exploring transfer learning for different domain adaptation. However, despite the transfer learning capability, it still requires both categories of images (real and fake). The common drawback of all these methods is that enough real and fake face image data are required for training. However, since Deepfakes techniques are getting more and more sophisticated and diverse, it is difficult to collect a sufficient amount of data every time a new technique is introduced. Further, we need a generalized approach to detect new Deepfakes, relying mostly on real images.

One-class classification models are based on the assumption that all the observations only belong to one class, "normal"; the rest of the observations are considered as "anomalies". These types of problems usually belong to the anomaly detection domain. One-class Support Vector Machine (OC-SVM) which is a particular case of support vector machine that separates data points from the origin by learning a hyper-plane in a Reproducing Kernel Hilbert Space (RKHS) and maximizing RKHS distance, is one of the most popular unsupervised learning methods that can detect anomalies. However, the application of non-parametric OC-SVM to the detection of deepfakes can lead to a high error rate with many support vectors. Oza and Patel proposed One-class Convolutional Neural Network (OC-CNN). The main idea of OC-CNN is to use a zero centered Gaussian noise in the latent space as the negative class and train the network using the cross-entropy loss. Their core objective is to make all negative distributions close to the hyper-plane.[3]

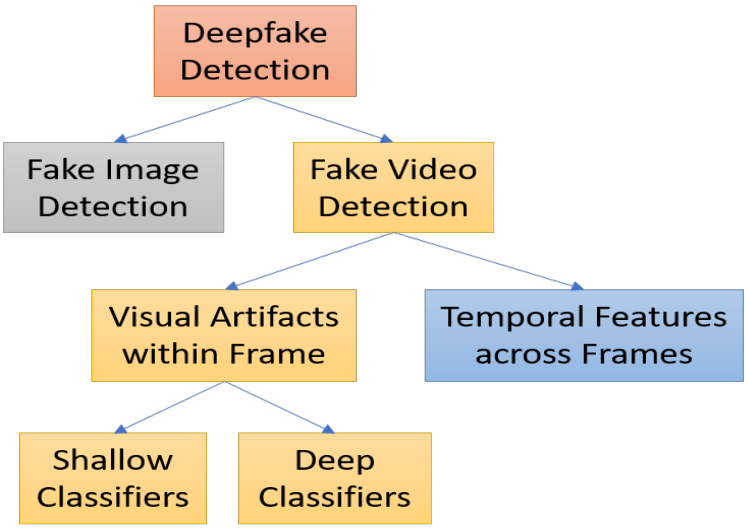


Figure 3.3: Deepfake Detection

3.1 MAJOR CONTRIBUTION

In this paper, we emphasize OC-based detection is indeed a viable option for the development of a generalized deepfakes detector, without the need for any fake images during the training phase. This cost effective method helps in preventing the usage of deepfakes in creating political distress,blackmailing, fake terrorism events, etc. and identity and privacy of person can be protected. The main contributions of this paper includes:

1. Investigating and evaluating the effectiveness of OC-based detection using anomaly score.
2. Comparing the results found to those reported in the literature

3.2 PROPOSED FRAMEWORK

1. **DATASET:** We used FaceForensics++ which is comprised of 5 types of deepfake data as well as normal data, as our baseline dataset. More than 1,000 YouTube videos have been collected and most videos present frontal faces without occlusions, which enables automated tampering methods for the generation of realistic forgeries. The authors of FaceForensics++ generate 4 different types of deepfake image using these videos, i.e., FaceSwap (FS), Face2Face (F2F), Deepfakes (DF) and NeuralTextures (NT). We used only the real images from this dataset to train our OC-FakeDect and both the real and fake images for testing.

   1.1 **Real Images:**

   FaceForensics++ offers a ground truth video dataset and applies different facial manipulation techniques to generate different deepfakes. We extracted face images from these original videos using Multitask cascaded Convolutional Neural Networks (MTCNN) and obtained 30,000 real human face images. These real images are used to train our OC-VAE.

   1.2 **FaceSwap dataset (FS):**

   .FaceSwap is a graphics-based approach to transfer a person's face from an image or a video to another. The face region is extracted based on sparse facial landmarks, which are used to fit a 3D template model using blend-shapes. This model is then projected back to the target image, and by using the textures of the input image, the difference between the projected shape and the localized landmarks

is minimized.

1.3 Face2Face dataset (F2F):

Face2Face is a real-time facial reenactment of a target video sequence. It animates the facial expressions of a target video from a source individual and re-renders the manipulated output video in a photo-realistic fashion, while maintaining the target person's identity. It is based on two input video streams with manual key frame selection, which are then used for the dense facial reconstruction and the re-synthesis of the face with different manipulations and expressions.

1.4 DeepFakes dataset (DF):

The DeepFakes dataset refers to the replacement of human faces using deep learning techniques. For the generation of forged videos, two Autoencoders sharing a single encoder trained to reconstruct the original and the target person's image, are used.

1.5 NeuralTextures dataset (NT)

NeuralTextures are learned feature maps of a target individual in a video. Originally, NT was trained with a photometric reconstruction loss with an adversarial loss, but as per the implementation by Rossler et al a patch-based GAN-loss is applied. To generate neural texture information, the tracking module of Face2Face is used to modify the mouth region.

1.6 Deepfake-detection dataset (DFD)

The Deepfakedetection dataset is provided by Google and JigSaw. It contains around 3,000 manipulated videos featuring 28 actors. Some paid actors were hired to record hundreds of videos for the generation of this dataset. From these videos, they created thousands of deepfakes using publicly available deepfakes generation methods. This dataset is now available as part of the FaceForensics++[10] benchmark dataset.

2. PRE PROCESSING

After collecting these real and fake video datasets, we extracted every frame from each video. Then, we used face detection and alignment using MTCNN[14] to extract human faces (from real and fake videos) and performed vertical alignment. We obtained 30,000 real human face images and 10,000 fake human face images for each dataset.

Figure 3.4: Examples of real and fake human face images extracted from the FaceForensics++ dataset. The first row contains real images, while other the rows below contain fake images of the DF, F2F, FS, NT and DFD datasets.

3.3 PROPOSED METHOD

OC-VAE is a Directed Probabilistic Graphical Model (DPGM) consisting of an encoder and a generator (decoder). In DPGM, graph expresses the conditional dependence structure between random variables. They are commonly used in probability theory, statistics particularly Bayesian statistics and machine learning. A VAE encodes the input as a distribution in the latent space. The objective function of VAE is

L(, , x) =DKL (q (z|x) || p (z)) Eq(z|x) (p (x|z)).

Here first term is the KL divergence of approximated posterior and the prior of the latent space. KL(Kullback-Leibler) divergence is the measure of how one probability distribution is different from the second reference probability distribution. The second term is calculated through the Monte Carlo method. Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. Overview of Monte Carlo method is that first we define a domain of possible inputs. Then we generate inputs randomly from a probability distribution over the domain. Then we perform a deterministic computation on the inputs and aggregate the results.

OC FakeDect's loss function is

$$L_{OC-FakeDect} = D_{KL}\left[N\left(\mu\left(x\right), \sigma\left(x\right)\right), N\left(0, I\right)\right] + \parallel X - p_{\theta}^{*}\left(Z\right) \parallel^{2}$$

Where X is the input and p∗0(Z)is the decoder output, μ (x) is the mean and (x) is the covariance. DKL is used to force the network to approximate a Gaussian distribution N (μ (x), (x)) in latent space, and mean square error to measure the difference between input and output image.

When sampling from the distribution returned by the encoder, the Monte Carlo gradient method is used to optimize the variational lower bound suffers from very high variance. So back propagation is not possible due to random sampling. To overcome this issue, latent space reparametrization technique is used. If Z is a random variable from a Gaussian distribution can be defined as follows:

Z = (x) × + μ (x), N (0, I)

- μ (x) : mean
- (x) : covariance

This equation ensures that the latent vector Z follows the posterior distribution, enabling us to train our model similar to training a VAE.

3.3.1 Anomaly Score

Anomaly score is also referred to as reconstruction loss or reconstruction score. To calculate the reconstruction score, we compute the Root Mean Squared Error (RMSE) between the input and an output image of VAE as follows:

$$rmse = \sqrt{\left(\frac{1}{n}\right)\sum_{i=1}^{n}\left(X_i' - X_i\right)^2},$$

- X : Original input
- X' : Reconstructed output

By computing this RMSE score for each image, one class distribution is constructed. Using that static threshold is determined to distinguish non-real images from real images.

3.3.2 Aritecture and Functioning

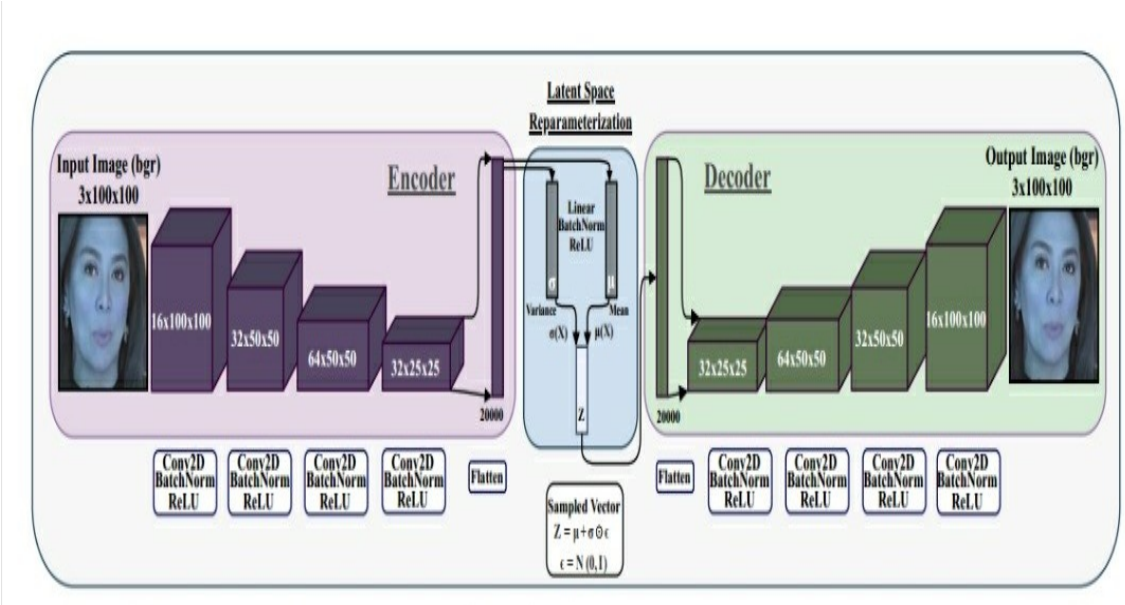


Figure 3.5: OC VAE Architecture

This is the architecture diagram of One-Class Variational Autoencoder. It consists of encoder and decoder. At encoder side image is given as input and we are scaling it at each stage. We are using convolutional layers and applying batch normalization with ReLU activation. A convolution is the simple application of a filter to an input that results in activation. Repeated application of same filter to input results in a map of activations called feature map. It indicates locations and strength of a detected feature in an input, such as an image. Batch normalization is applied inorder to speed up learning while training model. Batch normalization normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. Rectified Linear Activation function (ReLU) is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero.

A distribution is returned by this encoder and latent space reparameterization is done to that distribution i.e. we find mean and variance and Z is calculated. Z is given as input

to decoder and again we are applying convolutional layers and batch normalization. Thus reconstructed image is produced.

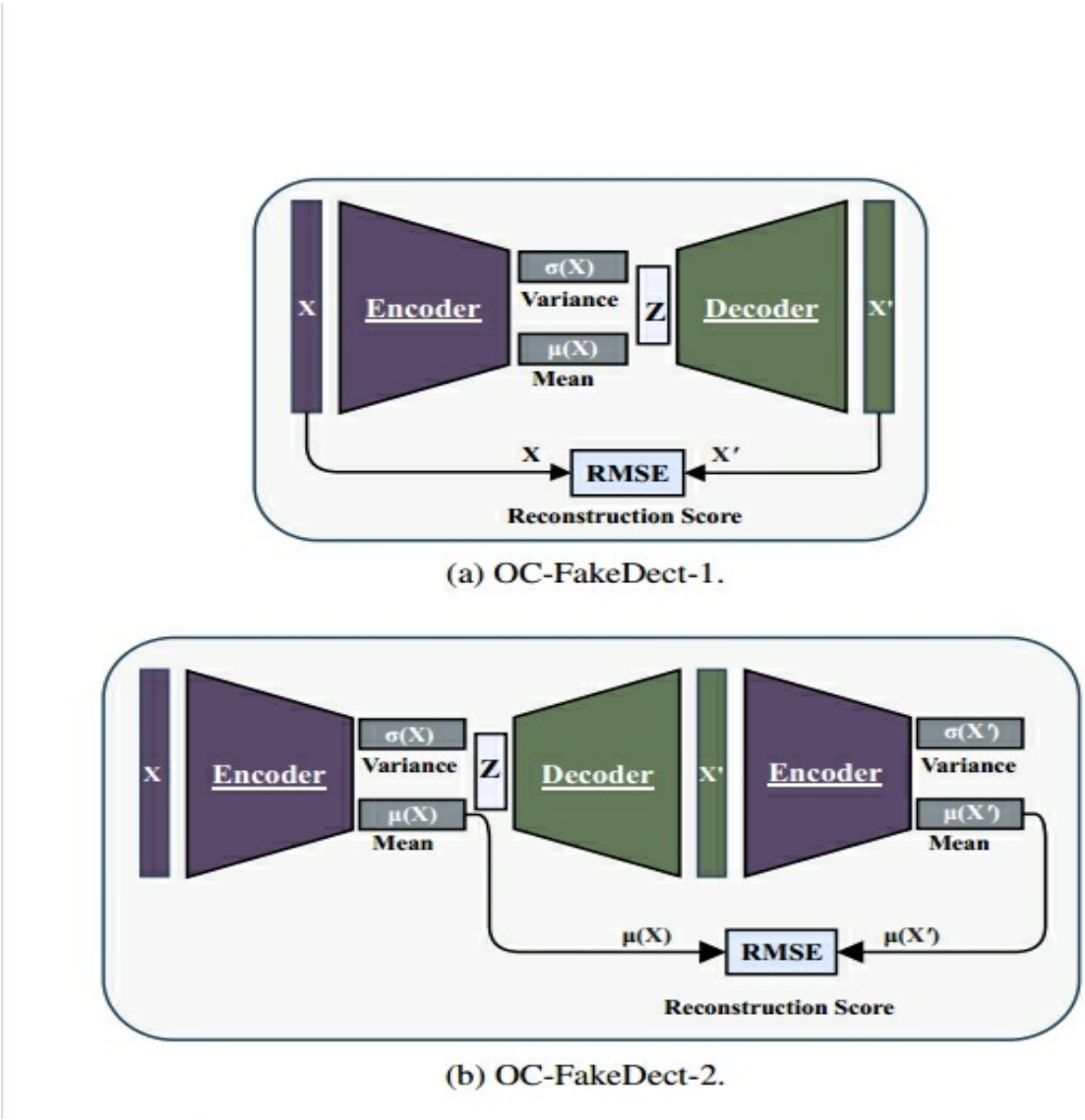


Figure 3.6: OC FakeDect Architecture

Based on the OC-VAE architecture as shown in figure, we propose two different OC-VAE-based approaches, OCFakeDect-1 and OC-FakeDect-2, to detect real and fake images. The first approach uses the same encoder and decoder building blocks, and loss function as OC VAC. In this first approach, we compute the reconstruction score by computing the RMSE between the original input X and the reconstructed output X0 to determine distinguish real and fake images. In the second approach, we have an additional encoder block following the decoder, which takes the decoder output X0 as an input and produces μ (X0). Then we compute the RMSE between the first encoder output μ (X) and the second encoder's output μ (X0) of the

input image. The additional encoder block can effectively extract real image features from the decoder output again, while lacking the ability to extract the features of non-real images such as deepfakes, giving high reconstruction score.

3.3.3 Histogram

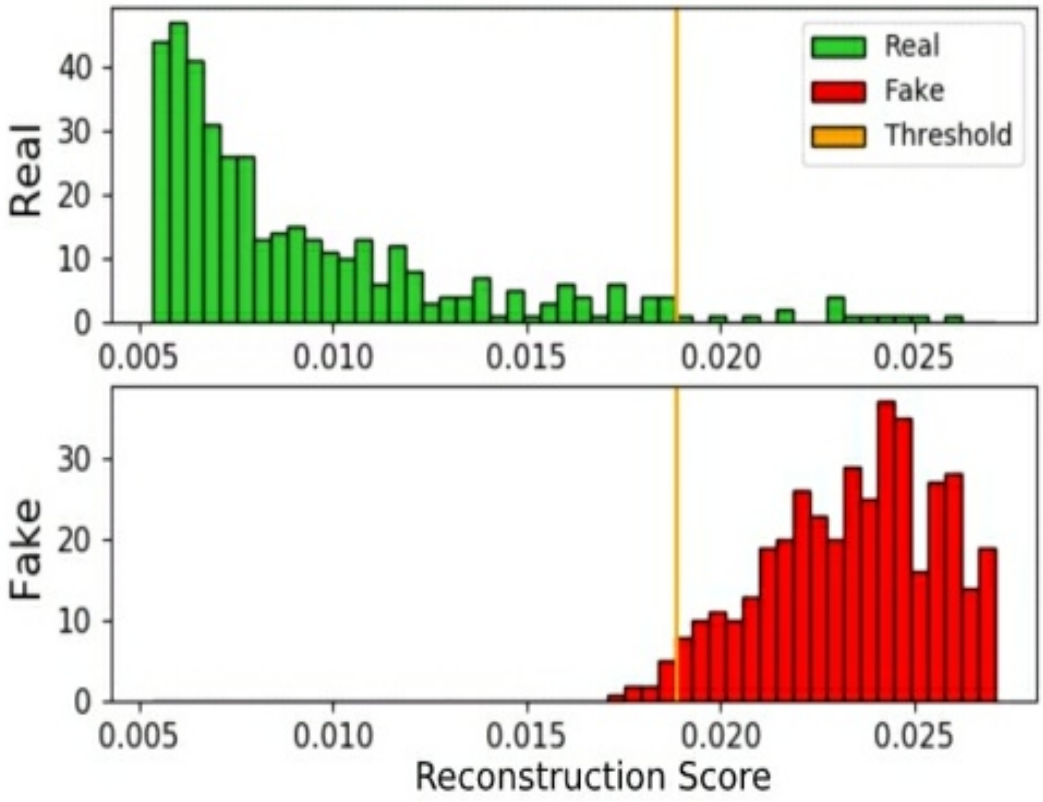


Figure 3.7: Histogram of reconstruction scores of real (green) and fake (red) images, and the statistical threshold (orange) on the NeuralTextures dataset with 50 real and 50 fake images.

When we construct histogram using resulting RMSE score, we get image as above. Here we can clearly identify the difference between real and fake images. If it is real image, its RMSE values will be low and if it is fake, RMSE values will be high.

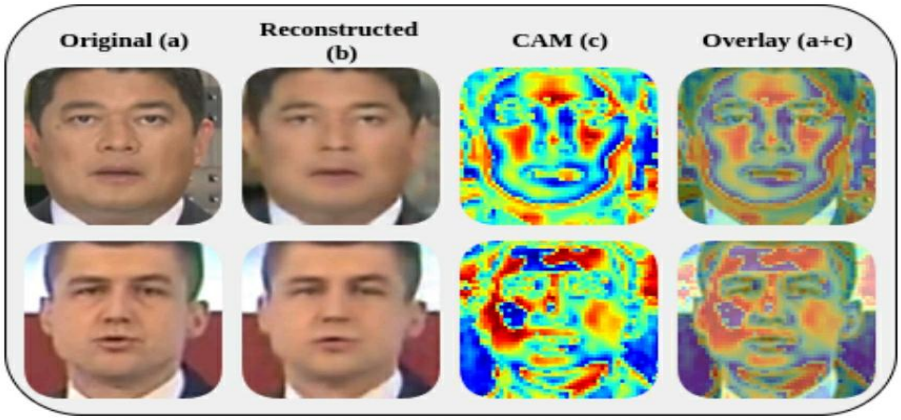
3.3.4 Data Augmentation

We applied various data augmentation techniques for the training and validation sets of real images, including horizontal and vertical flipping, and change of brightness, hue, and saturation with a factor of 0.05. Further, we normalized the distribution with a mean and standard deviation of 0.5. Examples of data augmentation techniques on the real image dataset from FaceForensics++ are shown in Figure 3.8.

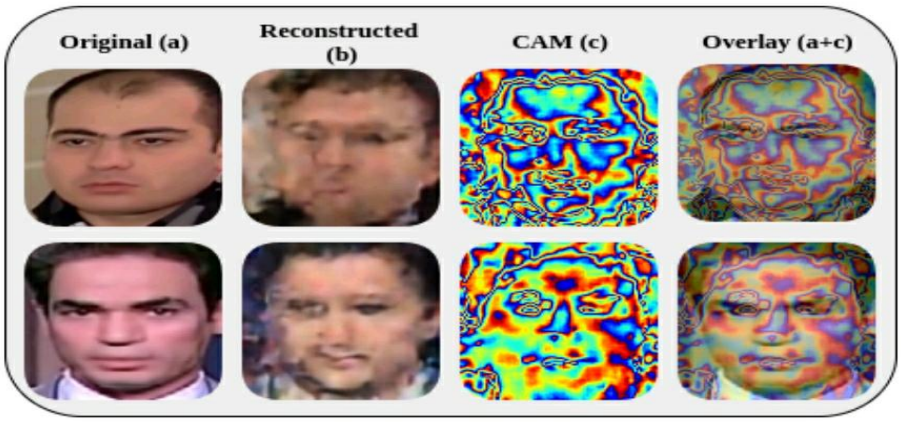
Figure 3.8: Data augmentation examples using real face images

3.3.5 Real vs Fake in OC FakeDect

Inorder to compare and visualize the learned features between real and faking images, we can employ GRADCAM (Gradient-weighted Class Activation Mapping)[11]. It highlights important region in the image for predicting the concept. Here we can clearly observe that OC-FakeDect produces intense activation around the face areas of real images, Face features such as nose, the forehead and the cheeks are clearly localized via class activation mapping. But in fake images it produces disordered and dispersed activation patterns. It complicates the perception of essential facial region compared to that in real images.



(a) Real face images.



(b) Fake face images.

Figure 3.9: : Class Activation Map (CAM) from OCFakeDect . The (a) original input, (b) the reconstructed output, (c) the CAM outputs, and (d) the overlaid images of the original input and its CAM for real and fake face images from NeuralTextures dataset are shown.

3.4 EXPERIMENTAL RESULTS

We scaled each image to 100 × 100 pixels and obtained 30,000 non-augmented real images, which will serve as the training data. For the training process of our OCFakeDect, we used the Adam gradient-based optimization method with a learning rate of 0.001 and batch-size of 128. We used convolutional layers and applied batch normalization with ReLU activation for each layer in both the encoder and the decoder. We trained our network for 300 epochs and chose the model yielding the best accuracy score on the validation set. We trained OC-FakeDect only with 30,000 real images, and without the fake images. For the testing phase, we used 500 real images and 500 fake images from the Deepfake, NeuralTexture, FaceSwap, Face2Face, and Deepfake Detection datasets, provided by the Face Forensics++ benchmark dataset.

For comparison, we used the one-class Autoencoder (OC-AE) as the baseline model, since it is widely used for OC classification tasks. We built an Autoencoder with three convolutional layers in the encoder and three convolutional layers in the decoder with batch-normalization and ReLU activation on every layer. Next, our proposed approaches are compared against each other, using the same training and testing procedure for all datasets. We compute the reconstruction score for each image and classify it as fake or real based on the threshold as described earlier.

3.4.1 Performance Table

Table 1: Performance of OC-AE, OC-FakeDect-1, and OC-FakeDect-2 on 5 different types of real and fake benchmark dataset provided by FaceForensics++. The threshold value is obtained based on the reconstruction score for each real and fake image of the testing data. The highest values are marked in bold and thresholds are underlined.

Dataset	Model	OC-AE (Baseline)			OC-FakeDect-1 (Ours)			OC-FakeDect-2 (Ours)		
	Type	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Deepfake	Real	0.492	0.492	0.492	0.860	0.864	0.862	0.885	0.882	**0.883**
	Fake	0.492	0.492	0.492	0.864	0.860	0.862	0.882	0.886	**0.884**
	Threshold			0.017			0.012			0.014
NeuralTexture	Real	0.547	0.548	0.548	0.952	0.954	0.953	0.979	0.970	**0.974**
	Fake	0.547	0.546	0.546	0.954	0.952	0.953	0.970	0.980	**0.975**
	Threshold			0.017			0.026			0.018
FaceSwap	Real	0.482	0.482	0.482	0.845	0.852	0.849	0.863	0.858	**0.860**
	Fake	0.482	0.482	0.482	0.851	0.844	0.847	0.858	0.864	**0.861**
	Threshold			0.017			0.010			0.012
Face2Face	Real	0.477	0.477	0.465	0.707	0.706	0.707	0.712	0.712	**0.712**
	Fake	0.477	0.477	0.475	0.707	0.708	0.707	0.712	0.712	**0.712**
	Threshold			0.017			0.006			0.009
Deepfake	Real	0.669	0.668	0.669	0.978	0.982	0.980	0.989	0.974	**0.981**
Detection	Fake	0.669	0.670	0.669	0.982	0.978	0.980	0.974	0.990	**0.982**
	Threshold			0.021			0.038			0.022

Figure 3.10: Performance Table

This is the performance table that we will get during testing phase. Precision, recall, and F1 score for both the real and fake datesets from 5 different sources in FaceForensics++ are calculated.

- Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.
- Recall: Recall is the ratio of correctly predicted positive observations to the all observa-

tions in the actual class.

- F1 score: F1 score is the weighted average of precision and recall.
- Threshold Value:Threshold value is a point beyond which there is a change.

Here we can see that OC FakeDect 2 returns highest F1 score than others. It means it is from OC FakeDect 2 we get more accurate results

Performance summary based on accuracy for all methods on the Deepfakes, NeuralTextures, FaceSwap, Face2Face and Deepfake-datasets

Model	NT	DF	FS	F2F	DFD
OC-AE	54.60	49.20	48.20	47.80	66.90
OC-FakeDect1	95.30	86.20	84.80	70.70	98.00
OC-FakeDect2	**97.50**	88.40	86.10	71.20	**98.20**
MesoNet [25]	40.67	87.27	61.17	56.20	N/A
Xcep. Net [25]	80.67	**96.36**	**90.29**	**86.86**	N/A

Figure 3.11: Performance Table

Here MesoNet and Xcep.Net are two class based methods. Here we can see OC-FakeDect scored highest accuracy in Neural Textures and Deepfake Dataset. But for Deepfake, Faceswap and Face2Face dataset Xcep.Net scored high accuracy, but the disadvantage is that it requires both dataset of real and fake for training. If enough dataset of fake images are not available for training, results won't be this much accurate. But in one class based methods, only real images are required for training and in those methods, OC FakeDect method scores highest accuracy.

CHAPTER 4
ADVANTAGES AND LIMITATIONS

4.1 Advantages

1. Need only real images for training
2. Cost effective
3. More accurate

4.2 Limitations

1. Better performance is only on the NT dataset and DFD dataset compared to XceptionNet, which is the current state-of-the-art method. So it needs performance improvement in other datasets as well
2. Here RMSE function is used to compute the reconstruction score of images. It would be better if we can develop a method in which the network itself provides a reconstruction score or develops a better anomaly scoring scheme

CHAPTER 5
CONCLUSION

In this paper, we formulate the challenging task of DeepFakes detection as a one-class problem using only real images for training. We propose OC-FakeDect , a model with a novel architecture, consisting of an additional encoder block to effectively learn the features of real images and detect anomalies, such as Deepfakes. Our proposed system outperforms other one-class-based approaches, as well as the two-class MesoNet. We also achieved higher performance on the NT dataset compared to XceptionNet. Using only the real images, our approach demonstrates that oneclass-based detection can be a promising option for coping with new or unseen deepfakes generation methods without the need for any of those fake samples. This cost effective method helps in preventing the usage of deepfakes in creating political distress,blackmailing, fake terrorism events, etc. and identity and privacy of person can be protected.

REFERENCES

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

[2] M. T. Jafar, M. Ababneh, M. Al-Zoube, and A. Elhassan. Forensics and analysis of deepfake videos. In *2020 11th International Conference on Information and Communication Systems (ICICS)*, pages 053–058, 2020.

[3] H. Khalid and S. S. Woo. Oc-fakedect: Classifying deepfakes using one-class variational autoencoder. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2794–2803, 2020.

[4] F. F. Kharbat, T. Elamsy, A. Mahmoud, and R. Abdullah. Image feature detectors for deepfake video detection. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–4, 2019.

[5] A. Kumar, A. Bhavsar, and R. Verma. Detecting deepfakes with metric learning. In *2020 8th International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6, 2020.

[6] S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *2011 International Conference on Computer Vision*, pages 2548–2555, 2011.

[7] A. A. Maksutov, V. O. Morozov, A. A. Lavrenov, and A. S. Smirnov. Methods of deepfake detection based on machine learning. In *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 408–411, 2020.

[8] Thanh Nguyen, Cuong M. Nguyen, Tien Nguyen, Thanh Duc, and Saeid Nahavandi. Deep learning for deepfakes creation and detection: A survey. 09 2019.

[9] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011.

[10] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner. Faceforensics++: Learning to detect manipulated facial images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11, 2019.

[11] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vis.*, 128(2):336–359, 2020.

[12] Q. Wang, J. Huang, Y. Feng, Z. Luo, C. Fan, and X. Liang. A comprehensive revisit to one class classification. In *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pages 337–344, 2017.

[13] M. A. Younus and T. M. Hasan. Effective and fast deepfake detection method based

on haar wavelet transform. In *2020 International Conference on Computer Science and Software Engineering (CSASE)*, pages 186–190, 2020.

[14] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[15] X. Zhou, H. Chen, L. Yan, Y. Xu, and X. He. Avae: Adversarial neural network for self-representation in one class classification. In *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, pages 1980–1984, 2019.