*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**

# DeepFake Detection Using Biological Method

*Submitted by*

**M P ADITHYA VIJAYAN**
**JEC17CS069**

*Supervised by*

**Ms. Aswathy Wilson**
**Assistant. 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**

# DeepFake Detection Using Biological Method

*Submitted by*

**M P ADITHYA VIJAYAN**
**JEC17CS069**

*Supervised by*

**Ms. Aswathy Wilson**
**Assistant. 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 **DeepFake Detection Using Biological Method** submitted by **M P ADITHYA VIJAYAN (JEC17CS069)** in partial fulfillment 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** | **Mr. Shaiju Paul** | **Dr. Vinith R** |
|---|---|---|
| Seminar Guide | Seminar Coordinator | Head of The Dept |
| Assistant Professor | Assistant 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 neural networks.
3. Analyzed and compared the general architecture of CNN,RNN and CRNN.
4. Studied about different DeepFake detection methods.
5. Analyzed the methodology of DeepFake detection using PPG signals.
6. Evaluated the above methodology for CNN, RNN and CRNN.

# 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. Our 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**, Assistant 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

In this busy growing internet age where many people are influenced by social media, it still remains a mystery to identify whether things on the web are real or not. DeepFakes also is known as AI videos which look real but are not actually real. This came into existence causing a major problem for the world like pornographic videos, political disputes, etc.

The proposed paper aims to put forward a technical solution to this problem by introducing a system that will detect and also identify if the video is real or not (DeepFake or not), using Deep Learning on PPG signals. This paper also proposes the evaluation and comparison of the performance of different PPG signals present in the video, thus concluding the effectiveness of using the most precise way for video detection and identification. The neural network which will be compared in this paper are CNN. This is also to validate the use of these deep neural network algorithms for DeepFake detection in real-life scenarios by using the biological methods and propose a DeepFake detection and identification system that would be better than the existing system of DeepFake detection and identification.

Keywords - Detection, Identification, Photoplethysmogram(PPG), Convolutional neural networks.

# CONTENTS

# List of Figures

# List of Abbreviations

| | |
|---|---|
| **CNN** | : *Convolutional Neural Network* |
| **RNN** | : *Recurrent Neural Network* |
| **CRNN** | : *Convolutional Recurrent Neural Network* |
| **ROI** | : *Region Of Interest* |
| **AI** | : *Artificial Intelligence* |
| **PPG** | : *Photoplethysmography* |
| **SVM** | : *Support Vector Machine* |

# CHAPTER 1
# INTRODUCTION

## 1.1 Overview

DeepFake videos are evolving rapidly and, not surprisingly, misuse of technologies are as well. In recent years, The DeepFakes which came in to existence for positive intent like movies,entertainment,advertisement and virtual clothing but later people started using it for negative intent. What makes them negative are they started using it in pornographic videos, Political speeches causes which made a catastrophe across the globe. This made the life of people much harder as the deeplearning and the advanced AI can help to make face and audio of people to make it look like real but are actually fake.


**Figure 1.1: PICTURE OF FIRST DEEP fAKE**

This paper proposes a deep fake source detection technique via interpreting residuals with biological signals. Biological signals are present in all humans. Anatomical actions such as heart beat, blood flow, or breathing, create subtle changes which are not visible to the eye

but still detectable computationally by other means. For example, when blood moves through the veins, it changes the skin reflectance(color) over time, due to the haemoglobin content in the blood. Approaches to extract photoplethysmography (PPG) signals are developed to recognize such changes by image processing techniques. The key observation is that biological signals are not yet preserved in deep fakes, and those signals produce different signatures for the generative noise. Thus, we can interpret biological signals as a projection of the residuals in a known dimension that we can explore to find the unique signature per model. This motivates us to utilize these signals for the recognition of the current and future generative models behind all deep fake videos. More importantly, the source detection can also improve the overall fake detection accuracy like to identify whether a video is fake or not. This includes occlusions, illumination changes,etc which can still be recognized as real videos as they do not conform to the signature of any generative model.The spatiotemporal patterns in biological signals can be conceived as a representative projection of residuals which can be used to classify a video and identify the specific generative model behind the deep fake.Pure deep learning based approaches try to classify deep fakes using CNNs where they actually learn the residuals of the generator these residuals contain more information and that can be used to reveal manipulation artifacts by disentangling them with biological signals.

## 1.2 Objective

The main objective of this seminar is to introduce a biological signal based detection and identification system which can distinguish between Real videos and DeepFakes using their PPG signals by deep learning. The use of this method had changed the identification of DeepFake to a new perspective.

## 1.3 Organization Of The Report

The report is organised as follow:

- **Chapter 1:Introduction** Gives an introduction of DeepFake and it's implementation.
- **Chapter 2:Literature Survey**Summarizes different DeepFake methods
- **Chapter 3: Deep Fake Source Detection Via Interpreting Residuals With Biological Signals**
- **Chapter 4:Conclusion** The overall conclusion for Different DeepFake
- **References** Includes the References for future Purpose

# CHAPTER 2
# LITERATURE SURVEY

## 2.1 Digital Forensics and Analysis of DeepFake Videos

Technological innovation especially in handheld devices with high quality cameras which are available on latest smartphones in combination with the spread of AI tools, models and apps have resulted in a huge number of videos of world famous celebrities and political leaders that have been used to mislead, To convey fake news for either political gains or in order to ridicule certain people which lead to catastrophe.With the growing of billions of digital images and videos daily across the several social media platforms together with the latest of deep learning apps allow us to create a fake video within a matter of minutes. Videos can be taken from huge repositories that are available on the internet, and with a few things in deep learning apps, it is very easy to create genuine-looking DeepFake videos. The potential impact of a fake video of a world leader can be a catastrophic and can have far-reaching implications for the world's economy and political stability of the country and can threaten the global peace, another side the use of these in generation of Pornographic videos.[7]

In order to prevent this,They used a deep-fake detection model with mouth features (DFT-MF), using deep learning approach to detect Deepfake videos by isolating, analyzing and verifying lip/mouth movement.They used CNN deep learning algorithm to classify deepfake videos. MoviePy, which an open-source application for editing and cutting videos, to cut the video based on certain words in which the mouth appears open and the teeth are visible. This is different from all other algorithms, that extract all images from the video and then attempt to identify the facial region within the extracted images.

1. **Creation Of DeepFake Videos**

   DeepFake is a video that has been constructed to make a person appear to say or do something that they never said or did which looks like real but are actually Fake. The first Deepfake was on early 2018,through the utilization of generative adversarial networks(GANs), which have led to the development of tools like Open Face, Swap Face2Face and Fake App that can generate videos from a large volume of images with only the requirement of minimum manual editing. Table 1 contains some of Common tools that Used to Create Deepfake videos. The main aim of deepfake algorithm is to allows

users to /transpose/replace the face of one person in a video with the face of another in a,realistic possible manner which is impossible to identify. To build deepfake videos, the deepfaker's need to assign two important GAN algorithm components,namely: the encoder network that will help to achieve a dimensional reduction by encoding the data starting from the input layer until it reduces the number of variables. The second one is the decoder network which reduces variables to create a new output very similar to the original as of illustrated in figure 1.

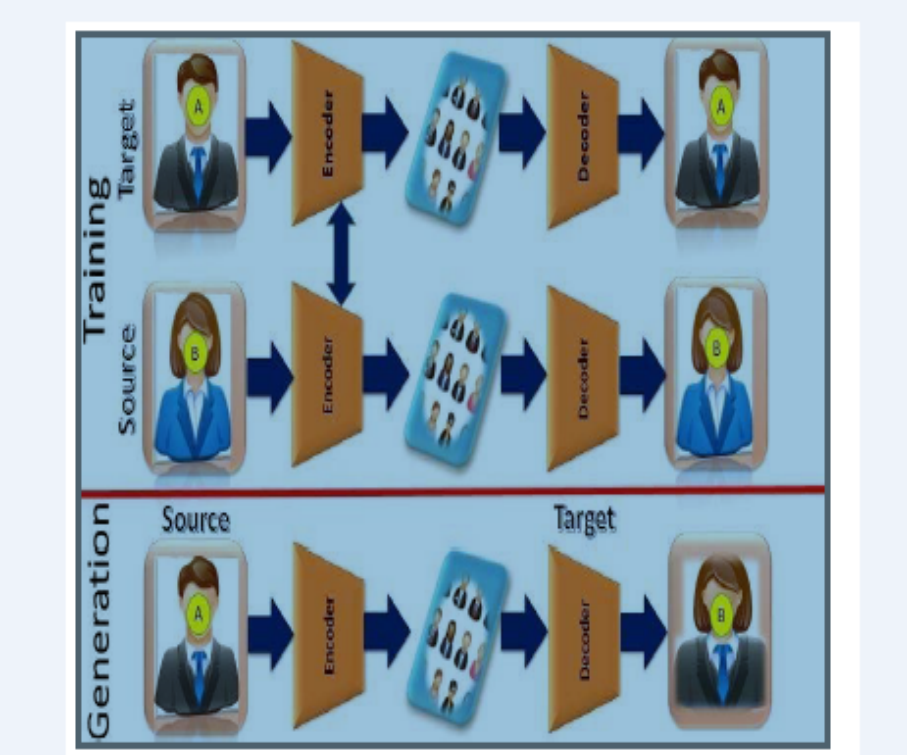


**Figure 2.1: Creation of DeepFake**

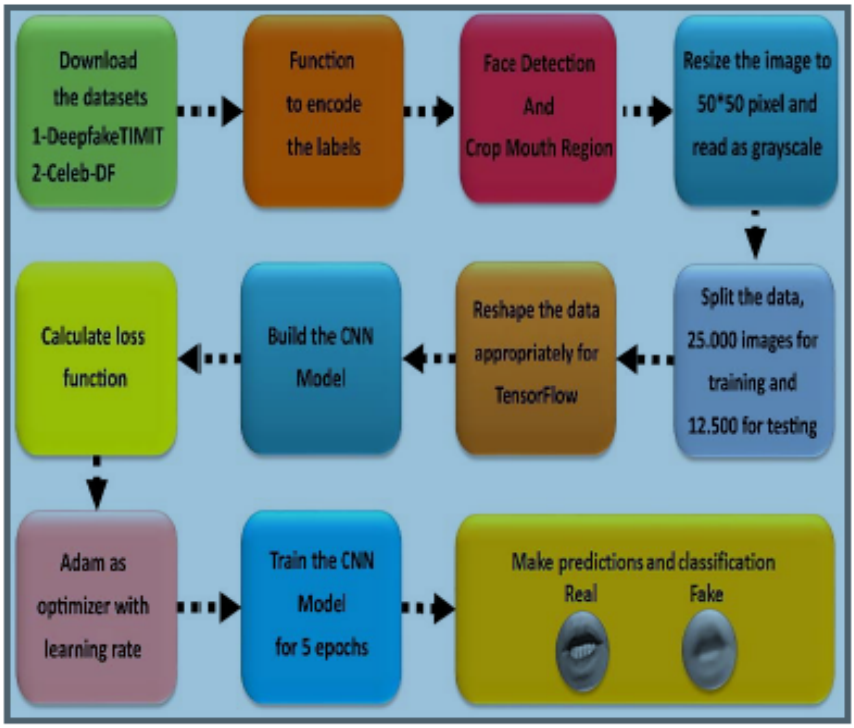


**Figure 2.2: Working steps**

- **Different Phases in DeepFake model**:

1. DataCollection

   In this stage, the information will be collected from relevant sources to create a new dataset that contains a combination of deepfake and real videos. They used the Deepfake Forensics (Celeb-DF) dataset and the Deepfake Vid-TIMIT dataset to extract all frames from the video before filtering irrelevant frames.

2. Pre-processing

   Prior to performing analysis on the image frames, some pre-processing is required. Face detection is one of the most essential steps of this work to enable us to filter out image frames (or parts thereof) that do not contain faces . TO this end, the Dlib classifier will be used to detect face landmarks and eliminate all unnecessary frames in the video.

3. Mouth Cropping

   The DFT-MF model focuses on area surrounding the mouth,especially the teeth; therefore, the mouth area will be croppedfrom a face in the frame. Working on a typical image frame ofa face, the facial landmark detector inside the Dlib

library is used to estimate the location of 68 (X, Y)-coordinates that graph to specific facial structures, these coordinates can be visualized as follows:

• The mouth can be located through points (49, 68).

• The right eyebrow through points (18, 22).

• The left eyebrow through points (23, 27).

• The right eye using points (37, 42).

• The left eye using points (43, 48).

• The nose is defined with points (28, 36).

• The jaw via points (1, 17).

The Dlib face landmark detector will return a shape object containing the face bounding box at 68 (x, y)-coordinates of the facial landmark regions in the image figure 3 illustrates that and at the points (49,68) the mouth can be located. 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 contains closed mouth by calculating distances between lips of the person. This is because of an image with a closed mouth has no fake value as nothing is being uttered in that frame. We will be tracking the open mouth, which the teeth with reasonable clarity so as to obtain high accuracy and increase efficiency of the model, figure 4 illustrates the idea.


**Figure 2.3: Face landmarks**

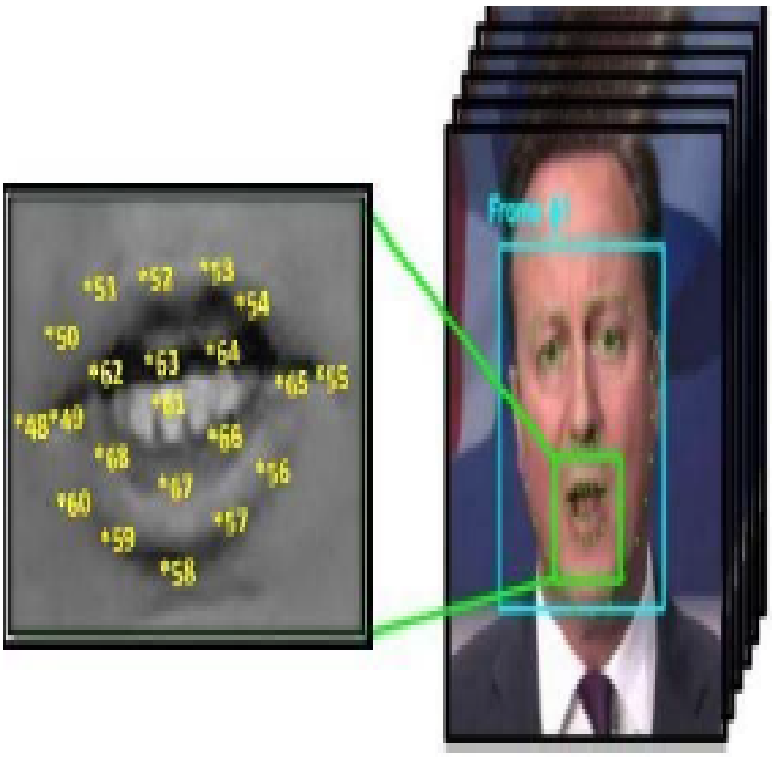


**Figure 2.4: Mouth Cropped**

4. DeepFake Video Classification

   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. As seen in figures

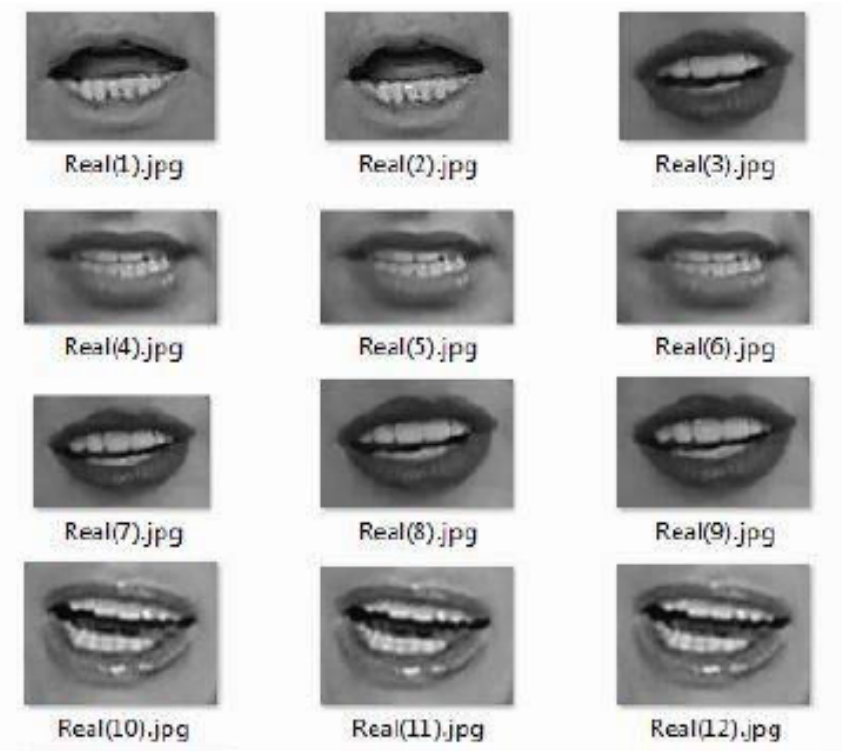


**Figure 2.5: Mouth Samples**


**Figure 2.6: Mouth samples with teeth**

The DFT-MF model uses AI concepts like deep learning,supervised learning ,Convolutional Neural Network (CNN), to classify videos to check whether video is fake or not 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.

Speech rate is used to describe how much words spoken per minute (wpm) in the video.

- **Conclusion**

They used CNN deep learning mechanism is used to classify deepfake videos using MoviePy for editing and cutting videos, to cut the video based on certain words in which the mouth appears open and the teeth are visible. They eliminated all images that containing closed mouth for easy identification.

The DFT-MF model was built to detect deepfake videos by using mouth as the biological signal.The datasets that were used contain both fake and real videos for the Celeb-DF and Deepfake-TIMIT then deep learning (CNN) was applied to classify fake videos depending on the features that will be taken from the mouth as a biological signal.

## 2.2 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.

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 uses linear, polynomial, radial basis function, and sigmoidal kernels, which provide the technique with the ability to solve different problems in different areas. 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.[8]

- 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. In general, the HOG process go into three phases for the divided blocks : (i) conduct optional global image normalization, (ii) computes the image gradients for both directions x and y, and (iii) use the overlapping local contrast normalizations to collect the HOG descriptors for all blocks.

- ORB

  The main advantage of ORB is that it is rotation invariant and resistant to noise and small changes .

- BRISK This method is used to detect corners and edges. BRISK is also rotation invariant and resistant to noise and small changes .

- 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

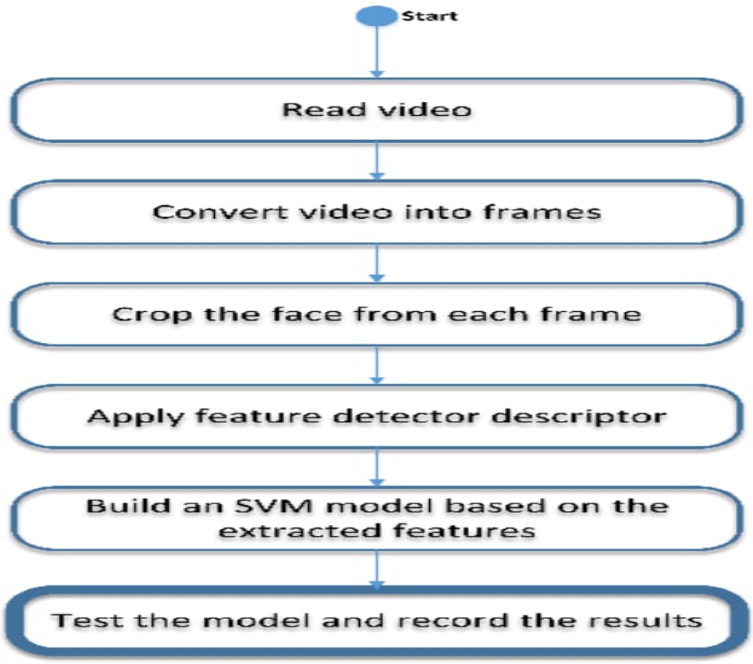


**Figure 2.7: Flow Diagram**

moderate increase in computational time".

- 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. We used (85%) and (15%) of the data set 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%. SURF and ORB also provides good accuracy exdeeding 90%. KAZE, on the other hand was the least effective with an accuracy of 76.5%. BRISK and FAST scores above 86.5% accuracy.

## 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 invideos that leaves few traces of manipulation, in what areknown 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. They used system u 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. We evaluate our method against a large set of deepfake videos collected from multiple video websites.Proposed system is composed of a convolutional LSTM structure for processing frame sequence 2 essential components in a convolutional LSTM:

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

CNN generates a set of features for each frame from an unseen test sequence Features of multiple consecutive frames are concatenated and passed on to the LSTM for analysis. Finally to estimate of the sequence video being either a deep or not.

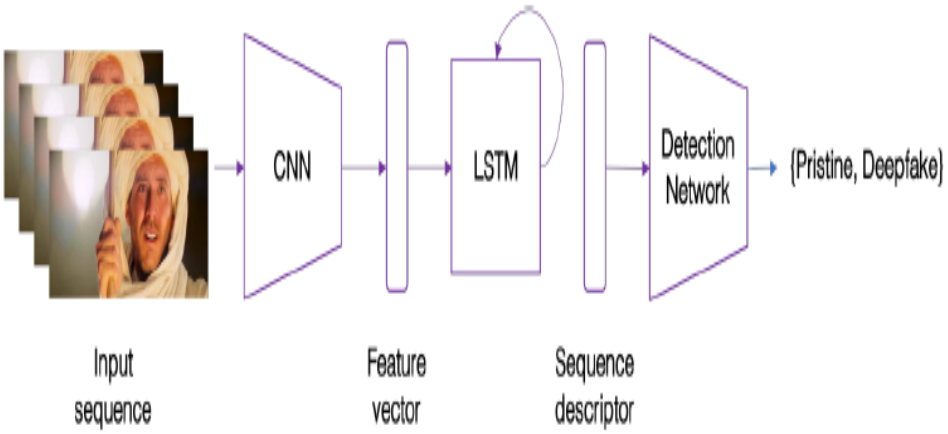


**Figure 2.8: Detection System**

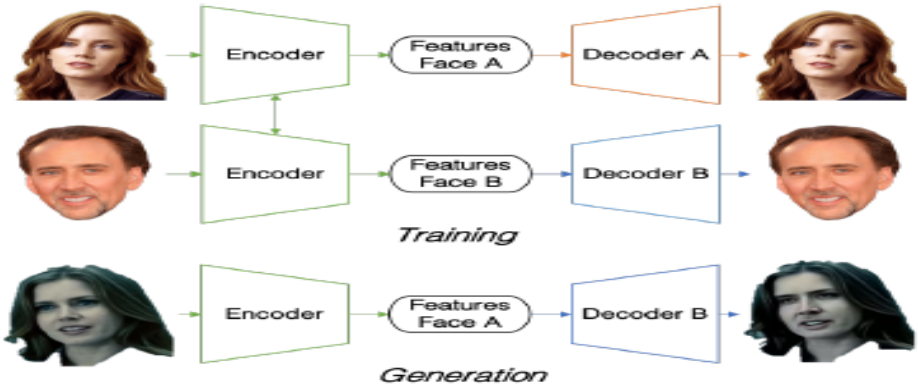


**Figure 2.9: Generation System**

### Architecture

Dataset is selected with 50% of deep fake videos and 50% of real videos. Random split is used to generate 3 disjoint set which is used for training, validation and testing. Balanced splitting on real videos and fake videos are done. This made sure that final set has exactly 50%of each class. Resize every frame to 299 X 299. Length of input sequence is controlled using subsequence sampling. This allows us to identify how many frames are necessary per video to have an accurate detection. Optimizer is set for end to end training of the complete model.

### Conclusion

In this paper they have presented a temporal aware system to automatically detect deepfake videos using LSTM within a fraction of 2 seconds.

## 2.4 DEEPFAKE DETECTION: CURRENT CHALLENGES AND NEXT STEPS

Current DeepFake detection methods mostly target faceswapping videos, which account for the majorities of DeepFake videos circulated online. Many of the existing methods are formulated as frame-level binary classification problems. Based on the features that are used, these methods fall into three major categories.[9] Methods in the first category are based on inconsistencies exhibited in the physical/physiological aspects in the DeepFake videos. The method in work of exploits the observation that many DeepFake videos lack reasonable eye blinking due to the use of online portraits as training data, which usually do not have closed eyes for aesthetic reasons. Incoherent head poses in DeepFake videos are utilized to expose DeepFake videos.The idiosyncratic behavioral patterns of a particular individual are captured by the time series of facial landmarks extracted from real videos are used to spot DeepFake videos[6]. The second category of DeepFake detection algorithms use signal-level artifacts introduced during the synthesis process.As synthesized faces are spliced into the original video frames, state-of-the-art DNN splicing detection methods,e.g.,[10] [1]can be applied. The third category of DeepFake detection methods are datadriven,which directly employ various types of DNNs trained on real and DeepFake videos but capturing specific artifact.

Mainly divided into 3 types

- Head puppetry In this the persons head is changed according to the situation keeping the rest of the body part as it is.
- Face swapping In this the face of the person is replaced by a Synthesized Face.
- Lip syncing In this lip region of the person is changed this is mostly done to say false things and create a problem.

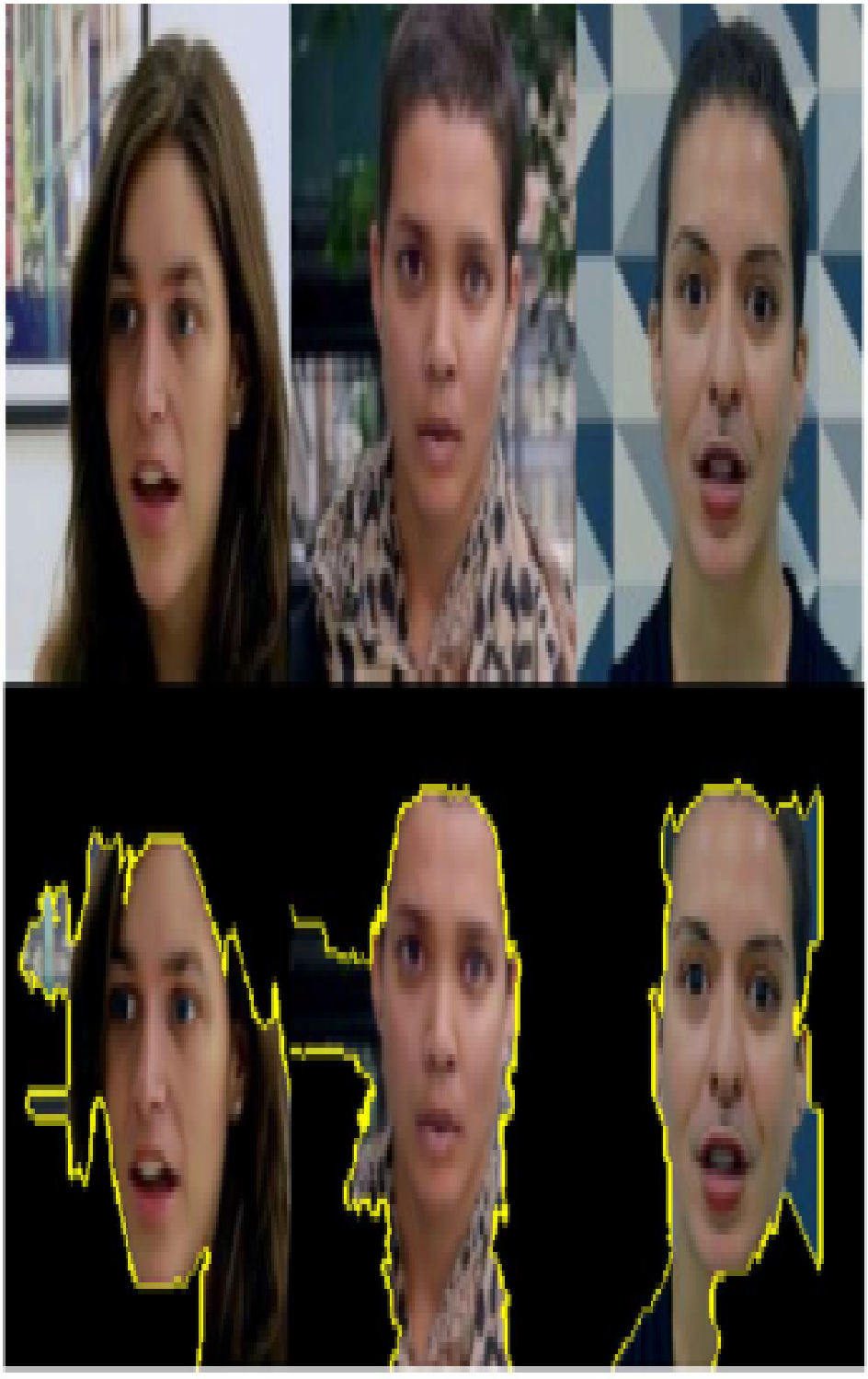

**Figure 2.10: Machine Identification Of DeepFakes**


**Figure 2.11: Example of DeepFakes**


**Figure 2.12: Different Types of methods used as above mentioned**

**Figure 2.13: Comparison with real face and Deepfakes**

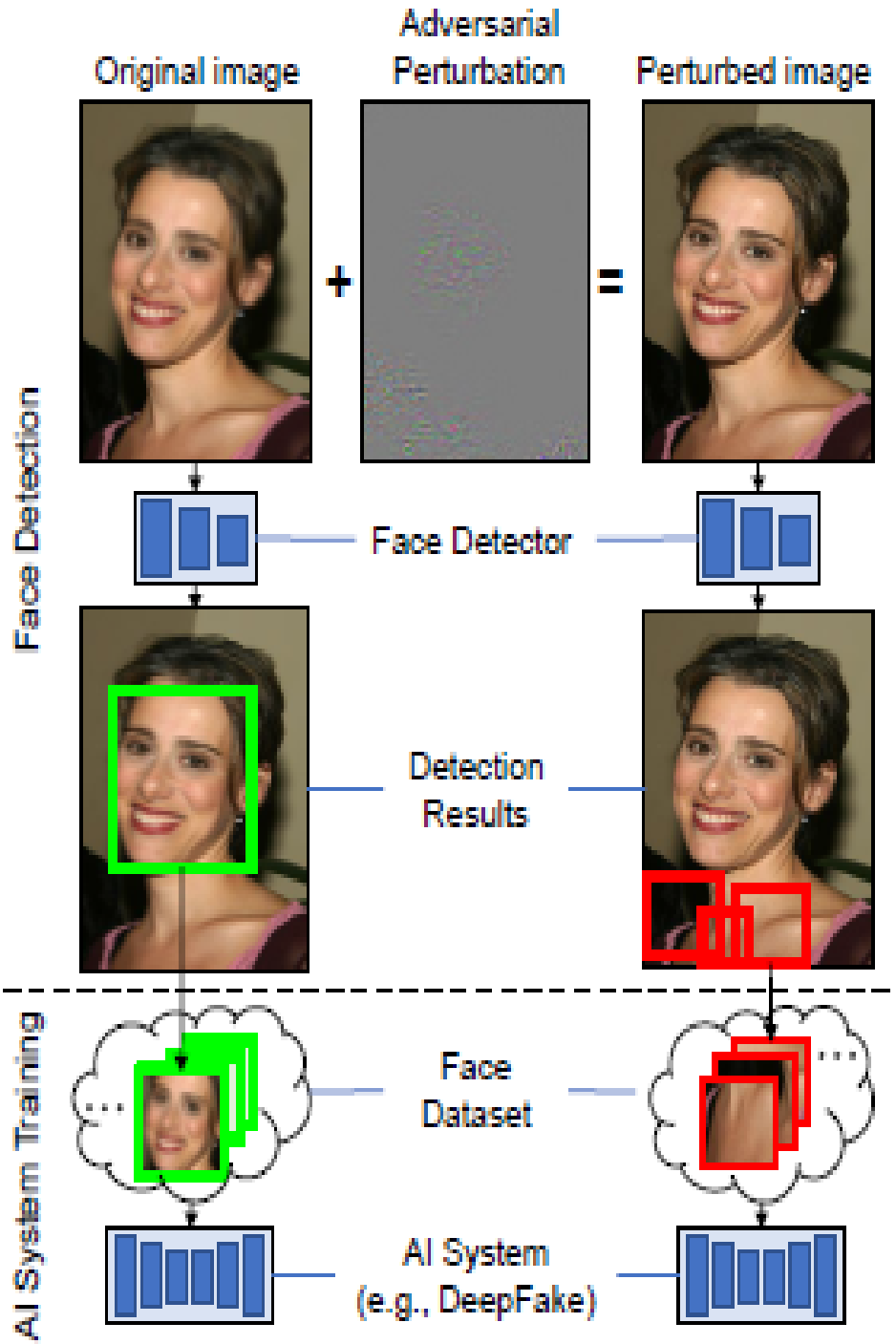


**Figure 2.14: Working**

# CHAPTER 3

# DEEP FAKE SOURCE DETECTION VIA INTERPRETING RESIDUALS WITH BIOLOGICAL SIGNALS

Deep Fakes are AI generated videos which look real but are actually fake.The development of Deep fake was meant for Positive Intent and people used it for Negative intent. The misuse of these led to so many catastrophe across the globe,These are developed using CNN(Convolutional Neural Network) in which the comparison takes place between real and fake videos. The Deep Fake was generally intended for positive use like in the famous movie fast and furious the recreation of Paul Walker after his death but people started using it for negative intendent like Pornography,Poltical issues etc. [4].

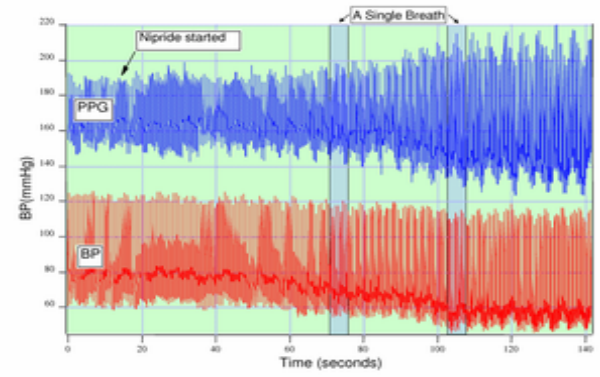


**Figure 3.1: PPG**

A photoplethysmogram (PPG) is an optically obtained plethysmogram that can be used to detect blood volume changes in the microvascular bed of tissue.

- **PPG and PPG Cells**

  Biological Signals are present in all Human beings.A photoplethysmogram ( PPG ) is an optically obtained plethysmogram that can be used to detect blood volume changes in the microvascular bed of tissue.Anatomical actions such as heartbeat ,blood flow,or breathing creating subtle changes that are not visible to the eye but still detectable computationally.

  In our present scenario there is no generative network for creating Deep fake with consistent PPG. A PPG is often obtained by using a pulse oximeter which illuminates the

skin and measures changes in light absorption; each person has a different signature with respect to the signals when measured against noise.[3]

They extracted 32 raw PPG signals from different locations in the face, from a window of frames, from a video of windows ,they then encode the signals along with their spectral density into a spatiotemporal block, which is so-called PPG cell.This cell is then fed into to the an off-the-shelf neural network to recognise the signatures of the distinct residuals of the source generative models.At last combine per sequence predictions into a per video prediction using average log of odds[2] .Their key finding emerges from the fact that we can interpret these biological signals as fake heart beats that contain a signature transformation of the residuals per model.

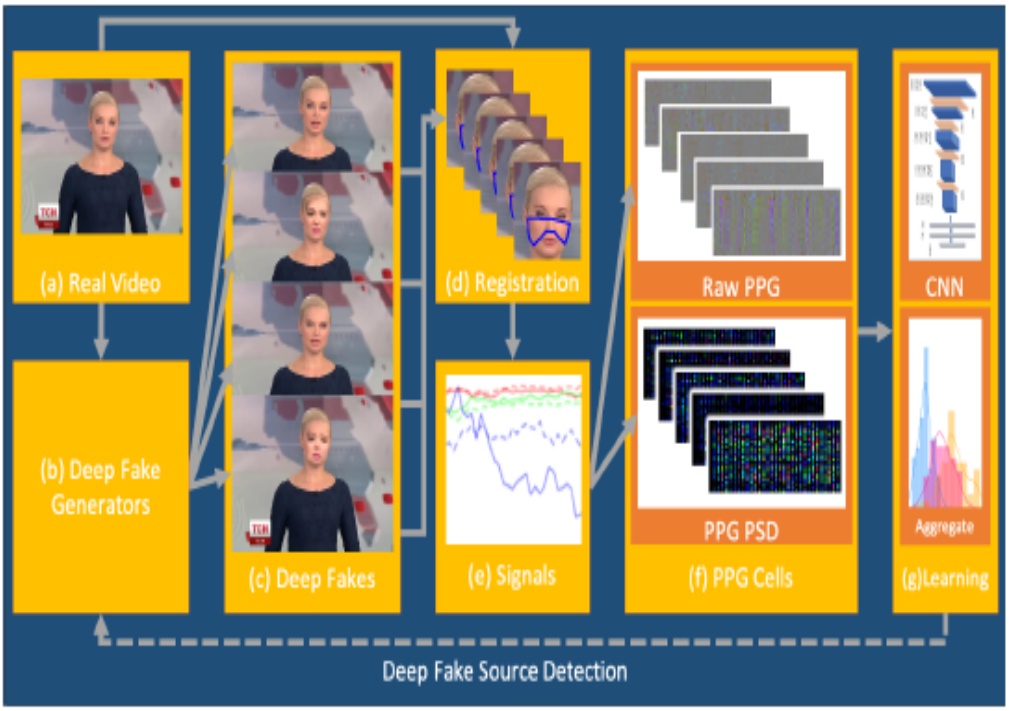


**Figure 3.2: Working Overview**

- **WORKING:**

1. **Defining PPG**:

   The first step is to capture the characteristics of biological signals consistently,for that define a novel spatio-temporalblock, called the PPG cell.The PPG cells combine several raw PPG signals and their power spectra, extracted from a fixed window. The generation of PPG cells starts with finding the face in every

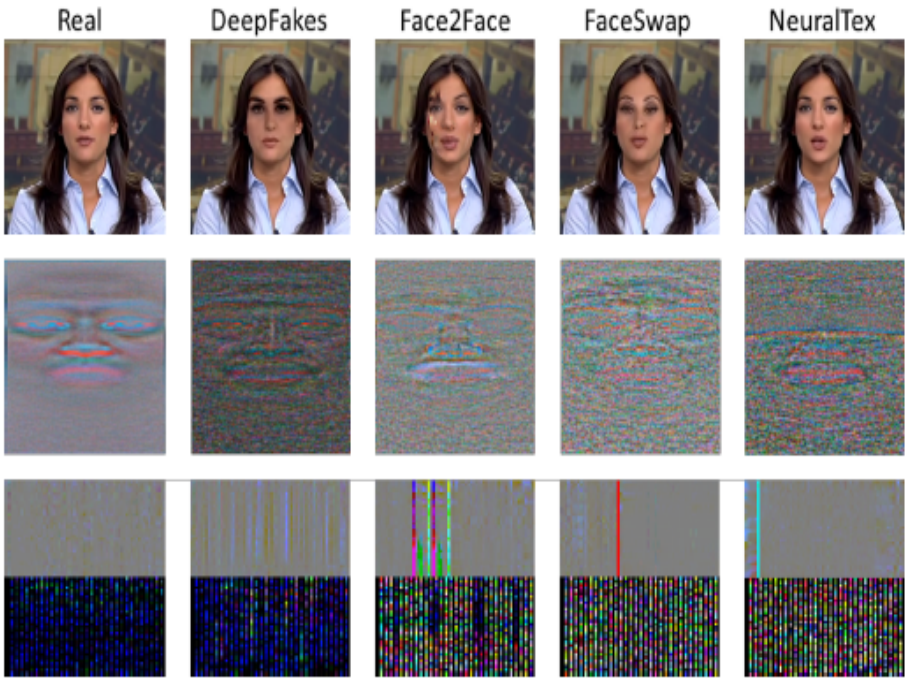


**Figure 3.3: PPG cells comparison**

frame using a face detector. In case the window contains multiple faces, we process signals individually and aggregate the results in the final step.

2. **Extraction of ROI**:

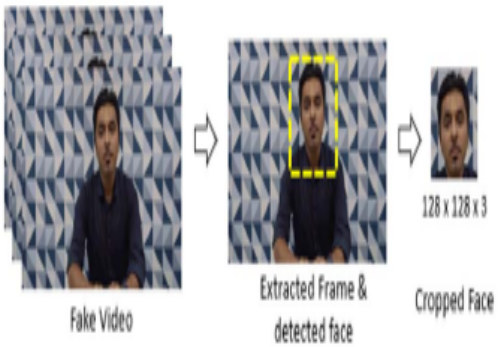


**Figure 3.4: Extraction of ROI**

The second step is to extract regions of interests (ROI)from the detected faces that have as much stable PPG signals as possible.Biological signals are sensitive to facial movements, illumination variations, and facial occlusions.In order to extract these areas robustly, They used the face region between eye and mouth re-

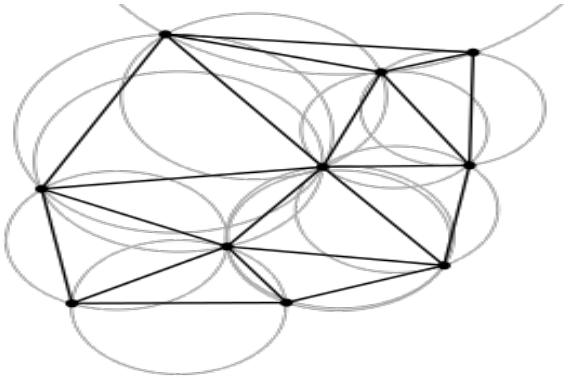

**Figure 3.5: Delaunay triangulation**

gions,maximizing the skin exposure,As the PPG signals from different face regions are correlated with each other, locating the ROIs and measuring their correlation become a crucial step to enhance the detection.

3. **ALIGNING OF ROI:**

   The third step involves aligning these nonlinear ROIs to a rectangular image,We employ Delaunay triangulation,followed by a nonlinear affine transformation per triangle to transform each triangle into the rectified image.[5]

4. **CALCULATING PPG**

   In the fourth step, They divided each image into 32 equal size squares and calculate the raw Chrom-PPG signal per square in a fixed window with the size of w frames, without interruptions in face detection Then, They calculated the Chrom-PPG in the rectified image since it produces more reliable PPG signals.Each window have w times 32 raw PPG values.They reorganized these into a matrix of 32 rows and w columns,forming the base of the PPG cells as shown in Figure top half of bottom rows .The bright columns correspond to significant motion or illumination changes where the PPG signal deviates abruptly.

5. **FINAL STEP:**

   The final step adds the information from the frequency domain to the PPG cells to calculate the power spectral density of each raw PPG value in the window and scale it to w size then concatenate the power spectra to the bottom to generate PPG cells with 64 rows and w columns.To analyze the contribution of the spectral information, we conduct experiments on PPG cells both with and without this last step and compare their accuracies.The projection of residuals of deep fake generators into the biological signal domain creates a unique pattern that can be utilized

for source detection.GAN residuals can be approximated by consistent noise in fake images.

Apply temporal non-local means denoising on the aligned face in one frame from each video in FF then accumulate and normalize the difference of original and denoised images, and subtract the noise of the real images from each corresponding fake residual to obtain the middle row in Figure containing the ”fingerprint” per generator.For the real class, demonstrate the overall noise accumulation.The colors of PPG-PSD correspond to different frequencies in the spectra of these residuals, and some of these frequencies are actually visible in the residual accumulation images. The main observation follows this correlation between the residuals and PPG cells: residuals create unique variations in the “deep fake heart beats” per model.

### 3.0.1 IMPLEMENTATION

The system is implemented in python utilizing Open-Face library for face detection, OpenCV for image processing, and Keras for neural network implementations ,Training and Testing NVIDIA GTX 1060 GPU with tractable training times .The most computationally expensive part of the system is the extraction of PPG cells from large datasets, which is a one time process per video.The process is to analysis, results, and some ablation studies. Unless otherwise noted, we set our testbed as the FF dataset with the same 70%-vs-30% split – 700 real videos and 4*700 deep fakes for training, and 300 real videos and 4*300 deep fakes for testing.

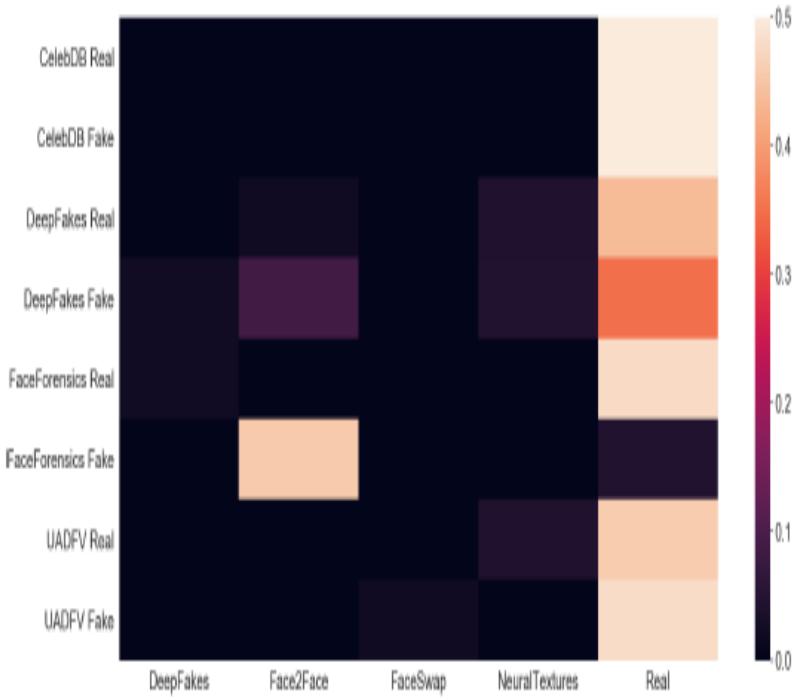


**Figure 3.6: Unseen dataset classification**

## 3.1 Extension to new models

1. **DATASET:** With respect to the data source the datasets can be of two types:

   (1) datasets with single model generation

   (2) datasets using multiple generative sources.

   some of the common examples for deepfake datasets are UADFV dataset contains 48 real and 48 fake videos generated by FakeAPP,DeepfakeTIMIT dataset have 650 deep fake videos generated using faceswap-GAN where as vidtimit videos are used as originals. FaceForensics dataset congregates 1,004 videos from the internet with their deep fake versions created by Face2Face, resulting in 2,008 videos. Celeb-DF dataset collects 590 real videos of famous actors, with 5,639 deep fake versions generated by an improved synthesis process A typical dataset generated by multiple generative methods is the commonly used FaceForensics++ (FF) dataset, which includes 1,000 real videos and 4,000 fake videos, generated by four generative models – FaceSwap, Face2Face, Deepfakes, and Neural Textures.

1.1 **DATA ACQUISITION:**

To acquire the PPG signals, the ROI of the videos source generated were recorded while in different background and lighting setup, this will enable us to publish the dataset publicly in order to be utilized by the research community while ensuring that no privacy or security regulations are being breached.

## 3.2 Neural Network- Convolutional Neural Network

Since CNN is more emphasized here, details of CNN is described below.

**Convolutional Neural Network:**

One of the most popular algorithm used in computer vision today is Convolutional Neural Network or CNN. Convolutional Neural Networks have a different architecture than regular Neural Networks. Regular Neural Networks transform an input by putting it through a series of hidden layers. Every layer is made up of a set of neurons, where each layer is fully connected to all neurons in the layer before. Finally, there is a last fully-connected layer — the output layer — that represent the predictions. Convolutional Neural Networks are a bit different. First of all, the layers are organised in 3 dimensions: width, height and depth. Further, the neurons in one layer do not connect to all the neurons in the next layer but only to a small region of it. Lastly, the final output will be reduced to a single vector of probability scores, organized along the depth dimension. CNN is composed of two major parts:

1. Feature Extraction: In this part, the network will perform a series of convolutions and pooling operations during which the features are detected. If you had a picture of a zebra, this is the part where the network would recognize its stripes, two ears, and four legs.

2. Classification: Here, the fully connected layers will serve as a classifier on top of these extracted features. They will assign a probability for the object on the image being what the algorithm predicts it is.

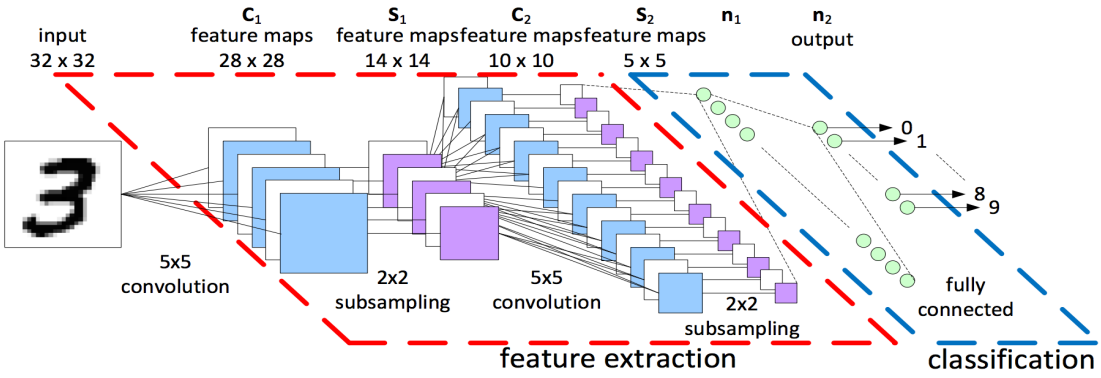


**Figure 3.7: Convolutional Neural Network**

There are squares and lines inside the red dotted region which we will break it down later. The green circles inside the blue dotted region named classification is the neural network or multi-layer perceptron which acts as a classifier. The inputs to this network come from the preceding part named feature extraction. Feature extraction is the part of CNN architecture from where this network derives its name. Convolution is the mathematical operation which is central to the efficacy of this algorithm. Lets understand on a high level what happens inside the red enclosed region. The input to the red region is the image which we want to classify and the output is a set of features.

### 3.2.1 Feature Extraction: Convolution

Convolution in CNN is performed on an input image using a filter or a kernel. To understand filtering and convolution you will have to scan the screen starting from top left to right and moving down a bit after covering the width of the screen and repeating the same process until you are done scanning the whole screen. For instance if the input image and the filter look like following: The filter (green) slides over the input image (blue) one pixel at a time starting from the top left. The filter multiplies its own values with the overlapping values of the image while sliding over it and adds all of them up to output a single value for each overlap until the entire image is traversed.

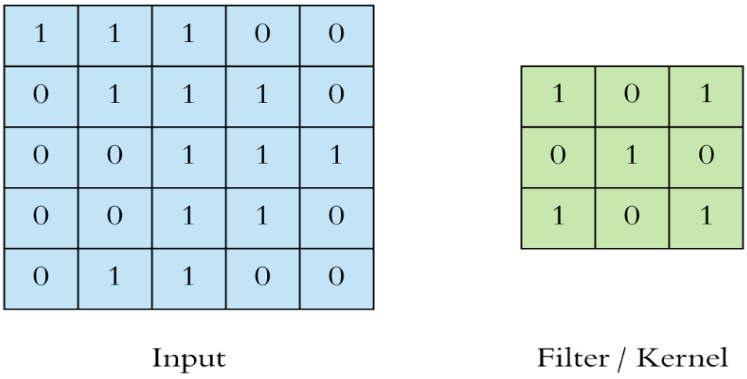


**Figure 3.8: Feature Extraction**

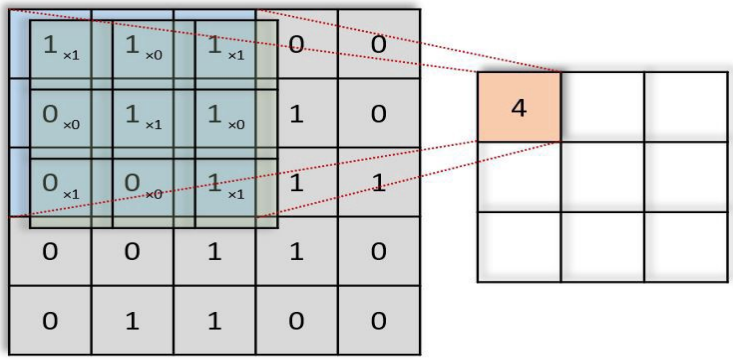


**Figure 3.9: Feature Extraction**

(1×1+0×1+1×1)+(0×0+1×1+1×0)+(1×0+0×0+1×1)=4 Similarly we compute the other values of the output matrix. Note that the top left value, which is 4, in the output matrix depends only on the 9 values (3x3) on the top left of the original image matrix. It does not change even if the rest of the values in the image change. This is the receptive field of this output value or neuron in our CNN. Each value in our output matrix is sensitive to only a particular region in our original image.

### 3.2.2 Feature Extraction: Padding

There are two types of results to the operation — one in which the convoluted feature is reduced in dimensionality as compared to the input, and the other in which the dimensionality is either increased or remains the same. This is done by applying Valid Padding or Same Padding in the case of the latter. In above example our padding is 1.

### 3.2.3 Feature Extraction: Example

Lets say we have a handwritten digit image like the one below. We want to extract out only the horizontal edges or lines from the image. We will use a filter or kernel which when convoluted with the original image dims out all those areas which do not have horizontal edges:

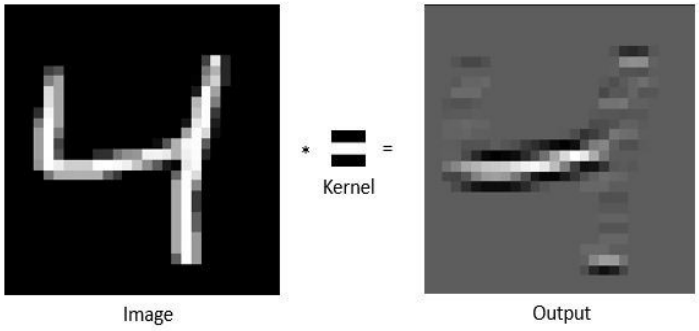


**Figure 3.10: Horizontal Filter**

Notice how the output image only has the horizontal white line and rest of the image is dimmed. The kernel here is like a peephole which is a horizontal slit. Similarly for a vertical edge extractor the filter is like a vertical slit peephole and the output would look like:

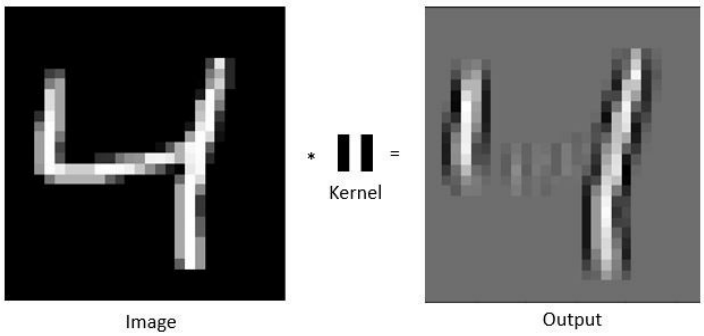


**Figure 3.11: Vertical Filter**

### 3.2.4 Feature Extraction: Non-Linearity

After sliding our filter over the original image the output which we get is passed through another mathematical function which is called an activation function. The activation function usually used in most cases in CNN feature extraction is ReLu which stands for Rectified Linear

Unit. Which simply converts all of the negative values to 0 and keeps the positive values the same:

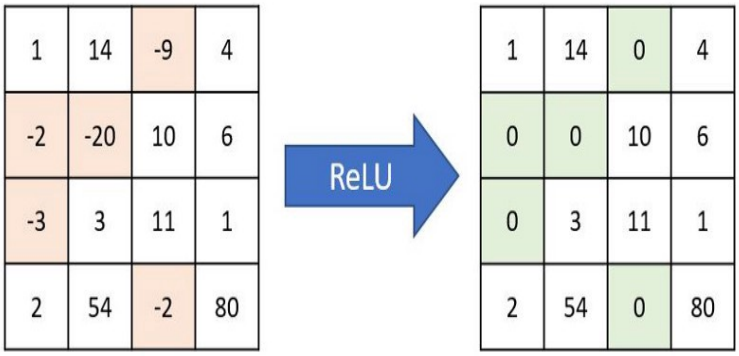


**Figure 3.12: Feature Extraction with ReLu**

After passing the outputs through ReLu functions they look like:

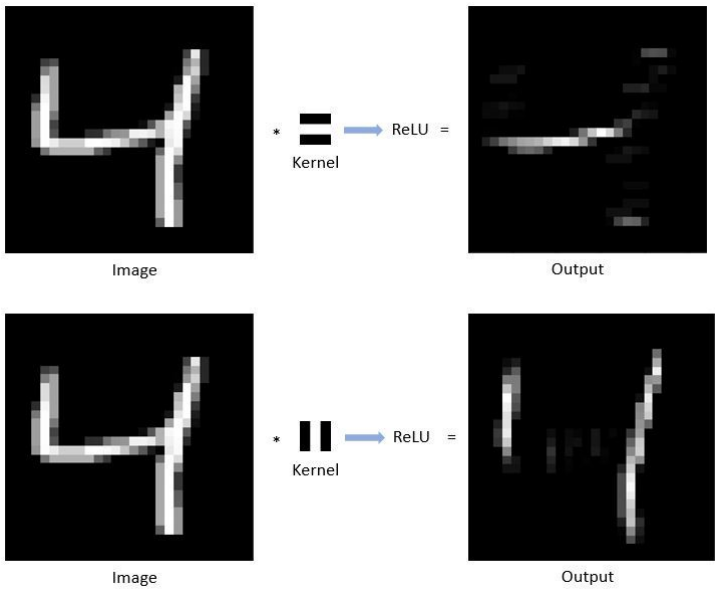


**Figure 3.13: Input after filtering with ReLu**

### 3.2.5 Feature Extraction: Pooling

After a convolution layer once you get the feature maps, it is common to add a pooling or a sub-sampling layer in CNN layers. Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant,

thus maintaining the process of effectively training of the model. Pooling shortens the training time and controls over-fitting. There are two types of Pooling- Max pool and Average pool. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

### 3.2.6 Classification — Fully Connected Layer (FC Layer):

Adding a Fully-Connected layer is a (usually) cheap way of learning non-linear combinations of the high-level features as represented by the output of the convolutional layer. The Fully-Connected layer is learning a possibly non-linear function in that space. Example of CNN network:

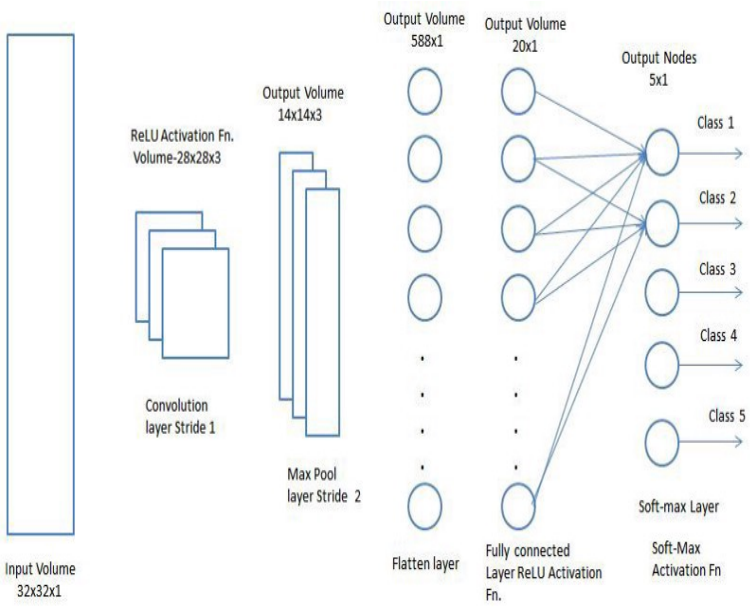


**Figure 3.14: Fully Connected Model**

Now that we have converted our input image into a suitable form, we shall flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training. Over a series of epochs, the model is able to distinguish between dominating and certain low-level features in images and classify them using the Softmax Classification technique.

## 3.3 Methodology

The three main objectives we are attempting to achieve are, firstly, we are interested in investigating the performance of the binary classification aspect on the problem and comparing the results with the literature. Secondly, observing and evaluating the outcome of the multi-class identification classification aspect of the problem and attempting to show the best algorithm out of the three algorithms implemented using different evaluation metrics. Finally, we aim to answer the question of which algorithm has the lowest training and evaluation time. In order to fulfill the three objectives above, different algorithms are compared and evaluated based on their accuracy, F1 score, precision and recall metrics. In order to understand the metrics, concept of confusion matrix needs to be explained.

### 3.3.1 Confusion Matrix

Confusion Matrix is a table with 4 different combinations of predicted and actual values.

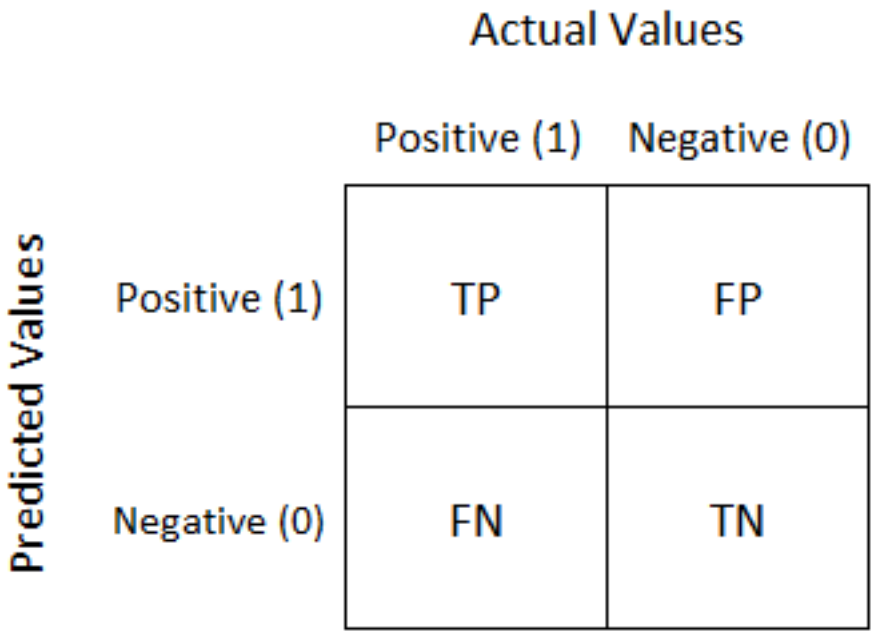


**Figure 3.15: Confusion Matrix**

1. **True Positive (TP):** A true positive is an outcome where the model correctly predicts the positive class.
2. **True Negative (TN):** A true negative is an outcome where the model correctly predicts the negative class.
3. **False Positive (FP):** A false positive is an outcome where the model incorrectly predicts the positive class.
4. **False Negative (FN):** A false negative is an outcome where the model incorrectly predicts the negative class.

### 3.3.2 Metrics

1. **Accuracy:** It is the ratio of number of correct predictions to the total number of input samples. It works well only if there are equal number of samples belonging to each class.

**Accuracy= (TP + TN)/(TP + TN + FP + FN)**

2. **Precision:** It is the number of correct positive results divided by the number of positive results predicted by the classifier.

**Precision= TP/(TP + FP)**

3. **Recall:** It is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive).

**Recall= TP/(TP + FN)**

4. **F1 Score:** F1 Score is used to measure a test's accuracy. F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances).

**F1 Score= (2*Precision*Recall)/(Precision + Recall)**

In order to ensure that every algorithm is performing at its optimum, we have carefully chosen the steps below to define the termination condition of the training phase:

1. Executing the algorithm with a very large number of training steps.
2. At an interval of 100 steps, the trained model is tested on the validation-set and the accuracy is calculated and recorded.
3. We compare the new accuracy of the validation-set with the best accuracy achieved so far.
4. If the accuracy did not improve over three successive validation tests, we test the trained model on the testing set and report the observed results.

## 3.4 Results

### 3.4.1 Source Classification Accuracy

To better evaluate our video source classification, we analyze how uniquely each generative model is detected using the biological signals as a modulator for residuals. This analysis supports our claim of different generative models having signature patterns projected to the biological signal space.

# CHAPTER 4
# CONCLUSION

Videos which look like real but are actually fake are Called as DeepFakes. DeepFake was developed for positive intent but people started using it for negative intent,PPG signals are used for identification of DeepFakes which gave a better result with accuracy this is the latest mechanism to find the DeepFake. ROIs are mainly used for the comparison between the videos RNN and CNN are used for the comparison of the data in the video frames to check the video has been manipulated or not.

# REFERENCES

[1] Jawadul H. Bappy, Cody Simons, Lakshmanan Nataraj, B. S. Manjunath, and Amit K. Roy-Chowdhury. Hybrid LSTM and Encoder-Decoder Architecture for Detection of Image Forgeries. *IEEE Transactions on Image Processing*, 28(7):3286–3300, July 2019.

[2] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face spoofing detection using colour texture analysis. *IEEE Transactions on Information Forensics and Security*, 11(8):1818–1830, 2016.

[3] U. A. Ciftci, I. Demir, and L. Yin. Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.

[4] Umur Aybars Ciftci, Ilke Demir, and Lijun Yin. How do the hearts of deep fakes beat? deep fake source detection via interpreting residuals with biological signals. *arXiv preprint arXiv:2008.11363*, 2020.

[5] Jacob E. Goodman and Joseph O'Rourke, editors. *Handbook of Discrete and Computational Geometry*. CRC Press, Inc., USA, 1997.

[6] 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.

[7] Mousa Tayseer Jafar, Mohammad Ababneh, Mohammad Al-Zoube, and Ammar Elhassan. Forensics and analysis of deepfake videos. In *2020 11th International Conference on Information and Communication Systems (ICICS)*, pages 053–058. IEEE, 2020.

[8] 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, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society.

[9] Siwei Lyu. Deepfake detection: Current challenges and next steps. *2020 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–6, 2020.

[10] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. Two-stream neural networks for tampered face detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1831–1839, 2017.