******

***Dissertation on***

**DeepFake Detection in Videos(Visual Part only)**

*Submitted in partial fulfilment of the requirements for the award of degree of*

**Bachelor of Technology**

**in**

**Computer Science & Engineering**

**UE20CS390A – Capstone Project Phase - 1**

***Submitted by:***

| **Thejas N U**  **Abbu Bucker Siddique**  **Mohammed Anas Danish**  **Hithesh Dinesh Nayak** | **PES1UG20CS606**  **PES1UG20CS617**  **PES1UG20CS614**  **PES1UG20CS646** |
| --- | --- |

*Under the guidance of*

| **Dr. Mamatha H. R.**  Designation |
| --- |

**January - May 2023**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

FACULTY OF ENGINEERING

**PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)

100ft Ring Road, Bengaluru – 560 085, Karnataka, India



**PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)

100 Feet Ring Road, Bengaluru – 560 085, Karnataka, India

**FACULTY OF ENGINEERING**

**CERTIFICATE**

*This is to certify that the dissertation entitled*

**‘Detection of deepfake videos’**

*is a bonafide work carried out by*

| **Thejas N U**  **Mohammed Anas Danish**  **Abbu Bucker Siddique**  **Hithesh D N** | **PES1UG20CS606**  **PES1UG20CS614**  **PES1UG20CS617**  **PES1UG20CS646** |
| --- | --- |

in partial fulfilment for the completion of sixth semester Capstone Project Phase - 1 (UE19CS390A) in the Program of Study - **Bachelor of Technology in Computer Science and Engineering** under rules and regulations of PES University, Bengaluru during the period Jan. 2022 – May. 2022. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The dissertation has been approved as it satisfies the 6th semester academic requirements in respect of project work.

| Signature  Dr. Mamatha H R  Professor | Signature  Dr. Shylaja S S  Chairperson | Signature  Dr. B K Keshavan  Dean of Faculty |
| --- | --- | --- |

**External Viva**

| **Name of the Examiners**  **1.** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Signature with Date**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| --- | --- |

**DECLARATION**

We hereby declare that the Capstone Project Phase - 1 entitled **“Detection of deepfake videos”** has been carried out by us under the guidance of Dr. Mamatha H R, Professor and submitted in partial fulfilment of the completion of sixth semester of **Bachelor of Technology** in **Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester January – May 2022. The matter embodied in this report has not been submitted to any other university or institution for the award of any degree.

| **PES1UG20CS606** | **Thejas N U** |  |
| --- | --- | --- |
| **PES1UG20CS614** | **Mohammed Anas Danish** |  |
| **PES1UG20CS617** | **Abbu Bucker Siddique** |  |
| **PES1UG20CS646** | **Hithesh D N** |  |

**ACKNOWLEDGEMENT**

We would like to express our gratitude to Dr.Mamatha H R, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE20CS390A - Capstone Project Phase – 1.

We are grateful to the project coordinator, Dr. Priyanka H., all the panel members & the supporting staff for organizing, managing, and helping the entire process.

We take this opportunity to thank Dr. Shylaja S S, Chairperson, Department of Computer Science and Engineering, PES University, for all the knowledge and support we have received from her.

We are grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, Dr. B.K. Keshavan, Dean of Faculty, PES University for providing us various opportunities and enlightenment during every step of the way.

Finally, this project could not have been completed without the continual support and encouragement we have received from our family and friends.

**ABSTRACT**

Deepfake videos have emerged as a major threat to cybersecurity and digital forensics in recent years, with the potential to cause significant harm by manipulating public opinion, spreading fake news, and damaging the reputation of individuals and organizations. To combat this threat, researchers and practitioners have developed various techniques and algorithms for deepfake detection.

Through a thorough literature review, this report finds that deepfake detection is a challenging problem that requires a multidisciplinary approach combining computer vision, machine learning, and forensic analysis. While current deepfake detection techniques have shown promising results, they are still limited in their ability to detect highly realistic and sophisticated deepfakes. Moreover, the effectiveness of these techniques may depend on the specific characteristics of the deepfake and the context in which it is deployed.

We aim to develop a DeepFake Detection model that can differentiate artificially created videos using deep learning from the real videos. This model uses a combination of deep learning techniques to process the input video and based on the data extracted, it will classify the video either as deepfake or real. This model can be useful in various fields such as social media, news platforms, news channels, etc., where they can verify if the video is real before it reaches the people and starts spreading virally.

Our model will have 2 components, one for extracting features and data from the frames and another one for processing the extracted data using deep learning techniques. The deep learning component will classify the video as either deepfake or not based on the data given by the data extraction layer. Our model will be robust against image orientation, noise in the data and quality of the image.

**TABLE OF CONTENT**

| **Chapter No.** | **Title** | | **Page No.** | |
| --- | --- | --- | --- | --- |
|  | | **INTRODUCTION** | | 1 |
|  | | **PROBLEM STATEMENT** | | 4 |
|  | | **LITERATURE REVIEW** | | 6 |
|  | | 3.1 Background on Embeddings and Drug Repurposing | | 6 |
|  | | 3.1.1 Seq2seq Fingerprint: An Unsupervised Deep Molecular Embedding for Drug Discovery | | 6 |
|  | | 3.1.2 Predicting New Molecular Targets for Known Drug | | 7 |
|  | | 3.1.3 Drug Target Identification Using Side-Effect Similarity | | 7 |
|  | | 3.2 Generating Embeddings for SMILES | | 8 |
|  | | 3.2.1 SMILES2Vec: An Interpretable General-Purpose Deep Neural Network for Predicting Chemical Properties | | 8 |
|  | | 3.2.2 SPVec: A Word2vec-Inspired Feature Drug-Target Interaction Prediction | | 8 |
|  | | 3.2.3 Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition | | 9 |
|  | | 3.2.4 SWeeP: Representing Large Biological Sequences Datasets in Compact Vectors | | 9 |
|  | | 3.3 Generating Embeddings for Gene Expression | | 10 |
|  | | 3.3.1 Drug Repurposing Using Deep Embeddings of Gene Expression Profiles | | 10 |
|  | | 3.3.2 A Large-Scale Gene Expression Intensity-Based Similarity Metric for Drug Repositioning | | 11 |
|  | | 3.3.3 Gene2vec: Distributed Representation of Genes Based on Co-Expression | | 11 |
|  | | 3.3.4 Deep Learning Applications for Predicting Pharmacological Properties of Drugs and Drug Repurposing Using Transcriptomic Data | | 11 |
|  | | 3.3.5 Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics | | 12 |
|  | | 3.3.6 Drug Repositioning: A Machine-Learning Approach Through Data Integration | | 12 |
|  | | 3.4 Adverse Drug Reactions | | 13 |
|  | | 3.4.1 Systematic Drug Repositioning Based on Clinical Side- Effects | | 13 |
|  | | 3.4.2 Predicting ADR of Combined Medication from Heterogeneous Pharmacologic Databases | | 13 |
|  | | 3.4.3 Predicting Adverse Drug Reactions Through Interpretable Deep Learning Framework | | 14 |
|  | | 3.4.4 Detecting Potential Adverse Drug Reactions Using a Deep Neural Network Model | | 15 |
|  | | **DATA** | | 17 |
|  | | 4.1 Overview | | 17 |
|  | | 4.2 SMILES: Structural Indication of a Drug | | 17 |
|  | | 4.3 LINCS: Functional Indication of a Drug | | 18 |
|  | | 4.4 Combined Feature Set for ADR Detection | | 19 |
|  | | 4.5 Classification Systems | | 21 |
|  | | 4.5.1 ATC: Anatomical Therapeutic Chemical Classification | | 21 |
|  | | 4.5.2 SIDER: Side Effects Resource | | 23 |
|  | | 4.5.3 Adverse effects combined dataset | | 24 |
|  | | **SYSTEM REQUIREMENTS SPECIFICATION** | |  |
|  | | **SYSTEM DESIGN (detailed)** | |  |
|  | | **IMPLEMENTATION AND PSEUDOCODE (if applicable)** | |  |
|  | | **CONCLUSION OF CAPSTONE PROJECT PHASE-1** | |  |
|  | **PLAN OF WORK FOR CAPSTONE PROJECT PHASE-2** | | |  |
| **REFERENCE/ BIBLIOGRAPHY** | | | |  |
| [**APPENDIX A DEFINITIONS, ACRONYMS AND**](https://docs.google.com/document/d/1sxLzRaxGh0fTgzlXiH-5RgoxpmOZgtX9RD5OH3Dfigw/edit#heading%3Dh.qzpp4n2gssp1) **ABBREVIATIONS** | | | |  |
| [**APPENDIX B**](https://docs.google.com/document/d/1sxLzRaxGh0fTgzlXiH-5RgoxpmOZgtX9RD5OH3Dfigw/edit#heading%3Dh.mywen4vn97v4) **USER MANUAL (OPTIONAL)** | | | |  |

**LIST OF FIGURES**

| **Figure No.** | | **Title** | **Page No.** | |
| --- | --- | --- | --- | --- |
| Figure 1 | ADRs caused by drugs when used separately as compared to used together | | | 8 |
| Figure 2 | Representation of the L1000 Assay | | | 25 |
| Figure 3 | Combined feature set for ADR prediction | | | 27 |
| Figure 4 | Representation of the combine feature set | | | 27 |
| Figure 44 | Graph of testing and training losses | | | 85 |

**LIST OF TABLES**

| **Table No.** | | **Title** | **Page No.** | |
| --- | --- | --- | --- | --- |
| Table 1 | [Statistics of the L100 Assay](#_heading=h.3znysh7) | | | 23 |

**Problem Statement**

With the rapid advancements in deep learning and computer vision technologies, the creation and proliferation of deepfake videos have become more accessible than ever before. Deepfakes are synthetic videos or images that are created using machine learning algorithms, and they have the potential to cause significant harm by manipulating public opinion, spreading fake news, and damaging the reputation of individuals and organizations. As a result, the detection and prevention of deepfakes have become critical challenges in the field of cybersecurity and digital forensics.

Our aim is to develop a model that can effectively detect deepfake videos(only the Visual Part), even when they are realistic and sophisticated, while minimizing the false positives and false negatives, which are also robust against different deepfake evasion methods and is scalable to detect different types of deepfakes like face swapping, facial manipulations, identity swap, face reenactment, attribute manipulation, and entire face synthesis, etc.

**Literature Review**

**“An explainable deepfake detection framework on a novel unconstrained dataset”**

“Sherin Mathews, Shivangee Trivedi, Amanda House, Steve Povolny, Celeste Fralick”

2023

**Outline:**

The paper proposes a novel deepfake detection framework that uses deep learning techniques to detect deepfake videos. The proposed framework is evaluated on a new unconstrained dataset, and the results show that the framework outperforms existing state-of-the-art methods.

**Dataset:**

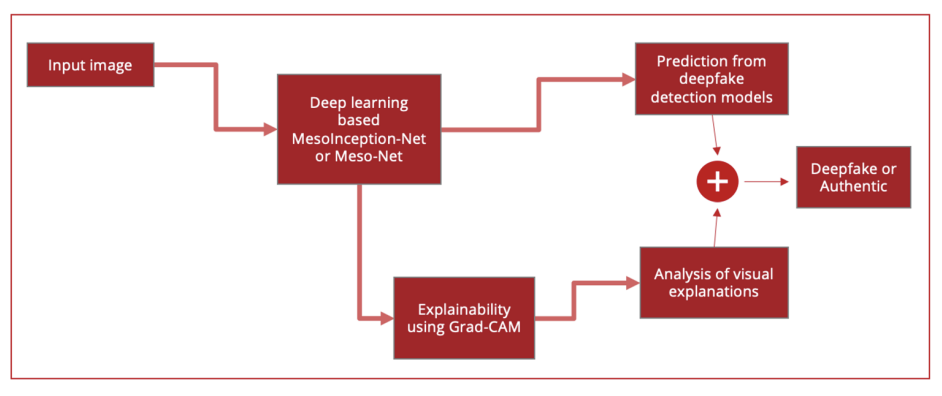
DeepFake Image-high-quality (DFIM-HQ) dataset which contains 70000 real images and 70000 fake images which have been created without any race,gender and age bias. It also includes various scenarios such as image orientation changes, low quality images and illumination degradations.

**Feature:**

The proposed framework uses a combination of handcrafted and learned features to detect deepfake videos. The handcrafted features include visual artifacts, such as unnatural skin tones and inconsistent head poses, while the features are learnt using Grad-CAM technique and the features will be trained on deep learning based model Meso-Net or MesoInception-Net

**Model Used:**

Meso-Net/ MesoInception-Net + Grad-CAM



**Gaps:**

* Mesonet and Mesoinception Net require a large amount of labeled data to be trained effectively. Acquiring and labeling data can be time-consuming and costly.
* Mesonet and Mesoinception Net have high computational complexity due to their multi-branch architecture, which can make them computationally expensive and slow to train and deploy.
* Mesonet and Mesoinception Net most of the times are vulnerable to adversarial attacks, where small perturbations to the input data can cause misclassification.
* The paper uses only two evaluation metrics (accuracy and F1-score) to evaluate the proposed framework's performance. Additional evaluation metrics could have been used to evaluate the framework's performance more comprehensively and provide a better understanding of its strengths and limitations

**Results:**

Accuracy:

Meso-Net:94.52%

MesoInception-Net: 99.87%

**“Multi-attentional Deepfake Detection”**

“Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, Nenghai Yu”

2021

**Outline:**

The paper proposes a novel multi-attentional deepfake detection method that leverages attention mechanisms to detect deepfake videos. The proposed method consists of two stages: feature extraction and classification. The feature extraction stage uses a deep convolutional neural network (CNN) to extract features from the input video frames, while the classification stage uses a multi-attentional mechanism to weigh the importance of different regions in the video frames for the final classification decision.

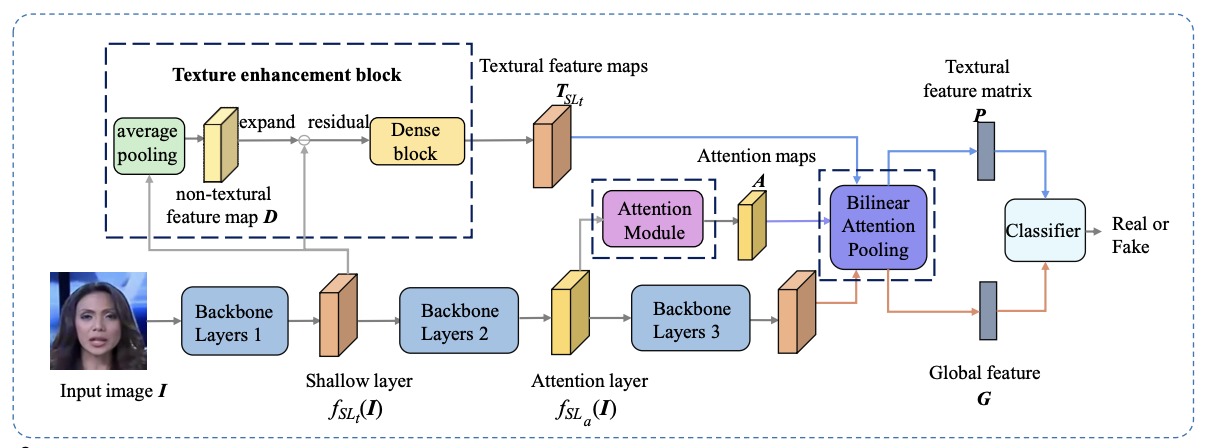
**Dataset:**

FaceForensics++, CelebDF and DFDC.

**Feature:**The proposed method uses a deep CNN to extract features from the input video frames. The CNN consists of multiple convolutional and pooling layers and is pre-trained on a large-scale face recognition dataset to capture robust facial features.

**Model Used:**

The proposed method uses a multi-attentional mechanism to weigh the importance of different regions in the video frames for the final classification decision. Specifically, the method uses three types of attention mechanisms: spatial attention, temporal attention, and channel attention. The three types of attention mechanisms work together to detect subtle inconsistencies and artifacts in the deepfake videos.



**Gaps:**

* The paper does not provide a detailed analysis of the complexity of the proposed model architecture.It would be helpful to have a more detailed analysis of the number of parameters, computation time, and memory requirements of the model. This analysis would help in understanding the scalability of the proposed model.
* It is unclear how the model performs when there are changes in the lighting conditions or camera angles in the input videos.

**Results:**

Accuracy:

Xception-net: LQ-86.95%, HQ-96.37%

Efficient-B4: LQ-88.69%, HQ-97.60%

**“Recurrent Convolutional Structures for Audio**

**Spoof and Video Deepfake Detection”**

“Akash Chintha , Bao Thai , Saniat Javid Sohrawardi , Kartavya Bhatt , Andrea Hickerson ,

Matthew Wright , and Raymond Ptucha”

2020

**Outline:**

This paper proposes a very systematic approach of deepfake detection Architecture of both visual part and Audio part of the Video.But according to the scope of our project we would be only considering the visual component of a video.

**Dataset:**

Face Forensics ++ , Celeb DF .

**Feature:**

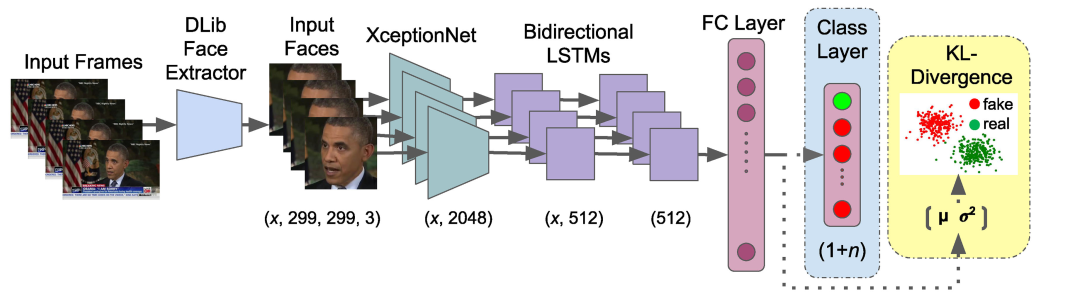
This paper uses a 3-Phase architecture where the first phase would be to use the Dlib library to take only the region of interest that is the face or body.

The second phase would be the neutral network which will use the weights and other parameters to represent the Image input to a latent representation.This includes the Xception Net which would be responsible for filtering only the spatial information and masking the temporal Information. then the output from that would be passed onto the 2-layered bi directional LSTM Model which would then generate the output which will be passed to the next phase.

The Third Phase would contain the outputLayer or the class layer.The Third phase will also contain the probability distributions of real and fake detections that have happened and then we have two models in the last phase so we have two loss functions which will be used one would be cross entropy between the real and fake output from neurons and other will be KL divergence loss for the probability distribution.

**Model Used:**

The models used are Xception Net for encoding and then Bi directional LSTM for the deep fake detection.



**Gaps:**

The model overfits for a comparatively smaller dataset.So we have to add any overfit prevention methods.

**Results:**

Accuracy would be as follows:

from the class layer the model gets the accuracy in range between 50 to 98.66

from the probability distribution we get the accuracy in range between 97 to 99

**“Detection of Deepfake Videos Using Long-Distance Attention”**

“Wei Lu , Lingyi Liu , Bolin Zhang , Junwei Luo , Xianfeng Zhao , Yicong Zhou , Jiwu Huang”

2023

**Outline:**

The paper proposes a new approach for detecting deepfake videos using long-distance attention mechanisms.The proposed approach focuses on analyzing the generated video frames for signs of manipulation using the long distance attention mechanism.It focuses on specific regions of video frames that are most likely to contain manipulations.

**Dataset:**

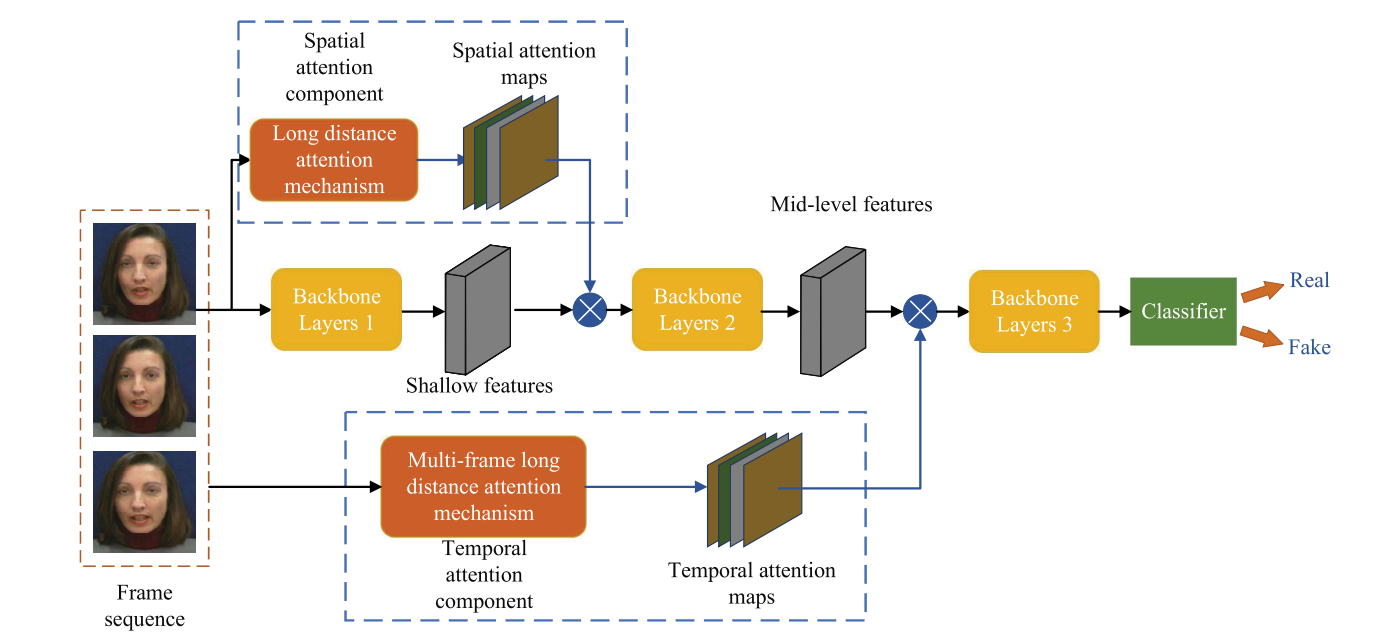
FaceForensics++, CelebDF

**Feature:**

It allows to focus on specific regions of the generated video frames that are most likely to have manipulations using the attention maps which is in contrast to the traditional attention mechanisms that only consider local regions of video frames.They have also used a combination of supervised and unsupervised learning techniques to train the built algorithm.

**Model Used:**

The proposed spatial-temporal model has two essential components.One is the spatial attention component and the other is the temporal attention component.



**Gaps :**

* Abnormal inputs such as incomplete faces, sideways face.
* Abnormal Processing of model such as generating abnormal attentions.

**Results:**

On CelebDF :

* ACC = 99.13
* AUC = 99.87

On FaceForensic++:

1. LQ
   1. ACC = 95.81
   2. AUC = 98.49
2. HQ
   1. ACC = 99.51
   2. AUC = 99.88

**“Fighting Deepfake by exposing the Convolution Traces on Images”**

“Guarnera, L., Giudice, O. and Battiato, S.”

2020

**Outline:**

The paper proposes an Expectation-Maximization algorithm which can be trained to detect and extract fingerprints left behind by GANs(Generative Adversarial Networks) called Convolutional Traces(CT). These Convolutional Traces can be used to not only to classify an images as deepfake or real but also help differentiate between the different GANs that have produced the deepfake image.

**Dataset:**

STYLEGAN, STYLEGAN2, FACEFORENSICS++

Also data was generated from the following models:

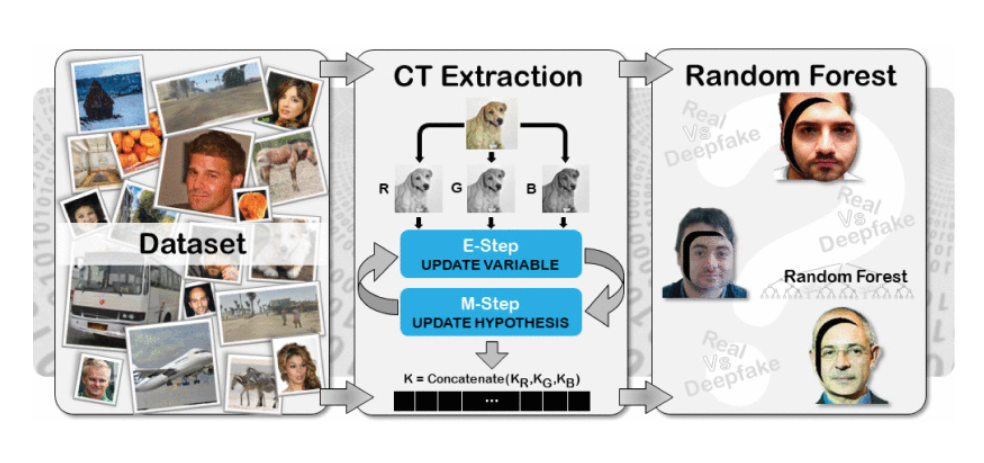
STARGAN, ATTGAN, GDWCT, CYCLEGAN, PROGAN, IMLE

**Feature:**

Since Expectation-Maximization method is not a deep learning method, It can easily be run on a common laptop without the need for high power GPUs. Moreover the extracted Convolutional Traces can also be used to identify which Generational Model was used to generate the given deepfake.

**Model Used:**

The author uses Expectation-Maximization to extract the CT(convolutional Trace) from the image which can then be passed through classifier such as random forest.

****

**Gaps:**

* Expectation-Maximization is very time consuming since it follows an iterative approach
* The model itself does not classify the images, rather it only extract the convolutional traces which then have to be passed through classifier. This adds to the execution time.

**Results:**

* Average accuracy of 87.1% over different classifier models
* Highest accuracy of 99.32% using random forest and **7x7** Kernel size on STYLEGAN2 dataset

**4. Data**

Deepfake datasets are collections of images, videos, and audio recordings that are used to train algorithms to create convincing fake media. These datasets typically consist of real media that has been manipulated using deep learning techniques to produce fake content. Deepfake datasets have become increasingly popular in recent years due to the growing concern about the potential misuse of this technology.

**4.1 DeepFake Detection Challenge**

The DeepFake Detection Challenge (DFDC) dataset is a large-scale dataset created by Facebook to aid in the development of deepfake detection technologies. The dataset consists of over 100,000 videos, both real and manipulated, and is divided into training, validation, and test sets. The manipulated videos were created using various deepfake creation methods, including GANs and autoencoders, to generate realistic fake media.

The dataset was created by collecting videos from publicly available sources, such as YouTube, and then manipulating them to create deepfakes. The manipulation process involved altering the faces of individuals in the video to create fake media. The manipulated videos were then labeled as "fake" and included in the dataset, along with the corresponding real videos.

The DFDC dataset includes a diverse range of subjects, lighting conditions, and camera angles to challenge deepfake detection algorithms. The dataset also includes a track for both image and video-based deepfakes, allowing researchers and developers to evaluate the performance of their deepfake detection algorithms on both types of media.

To evaluate the performance of deepfake detection algorithms on the dataset, Facebook organized a competition in which participants developed and submitted their algorithms for evaluation. The competition included a leaderboard that ranked the performance of each algorithm on the test set of the dataset. The DFDC dataset has spurred the development of new deepfake detection algorithms and helped researchers and developers to better understand the complexities of detecting manipulated videos.

**4.2 FaceForensics++**

The FaceForensics++ dataset is a large-scale dataset created to aid in the development of deepfake detection technologies. It contains 363 real and 3,068 deepfake videos. The manipulated videos were created using four different deepfake creation techniques: DeepFake, Face2Face, FaceSwap, and NeuralTextures.

The DeepFake method uses GANs to generate manipulated images, while the Face2Face method uses facial tracking to animate the expressions of one face onto another. The FaceSwap method uses deep learning algorithms to swap the faces of two individuals, and the NeuralTextures method uses GANs to synthesize facial features onto a target image.

The dataset includes multiple variations of each video, such as different compression levels and resolutions, to simulate real-world scenarios where deepfakes may be found. Each video is labeled as either real or manipulated, and the manipulated videos are further labeled with the creation method used.

In addition to the videos, the FaceForensics++ dataset also includes precomputed image features for each video, allowing for faster training and evaluation of deepfake detection models. The dataset has been widely used by researchers and developers to develop and test new deepfake detection methods and technologies.

**4.3 CelebDF**

The Celeb-DF(v2) dataset is a large-scale dataset created for the purpose of training and testing deepfake detection algorithms. It includes 590 videos which are collected from “Youtube” and 5639 synthesized deepfake videos. The manipulated videos were created using several deepfake creation methods, including DeepFake, FaceSwap, and Face2Face.

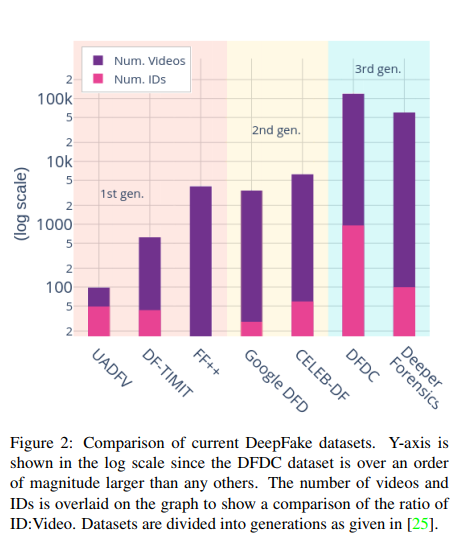
The DeepFake method uses GANs to generate manipulated images, while the FaceSwap method uses deep learning algorithms to swap the faces of two individuals, and the Face2Face method uses facial tracking to animate the expressions of one face onto another.

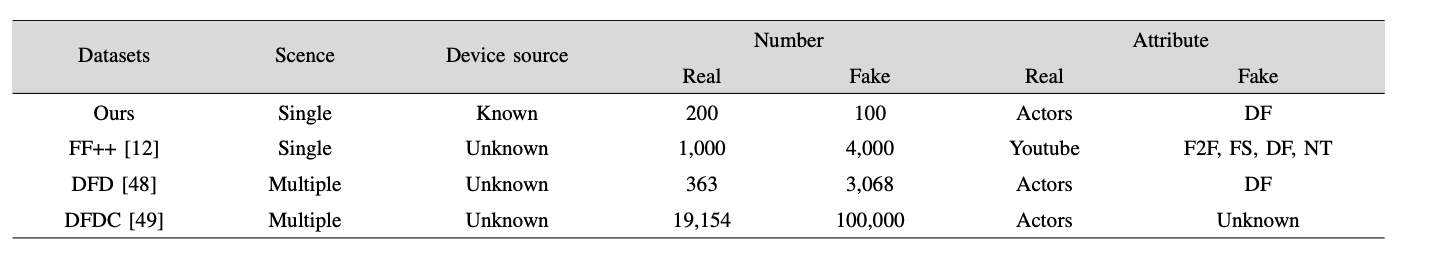
The Celeb-DF(v2) dataset was created by collecting videos of celebrities from publicly available sources, such as YouTube, and then manipulating them to create deepfakes. The dataset also includes various metadata for each video, such as the video's resolution and frame rate, to provide additional information for deepfake detection algorithms.

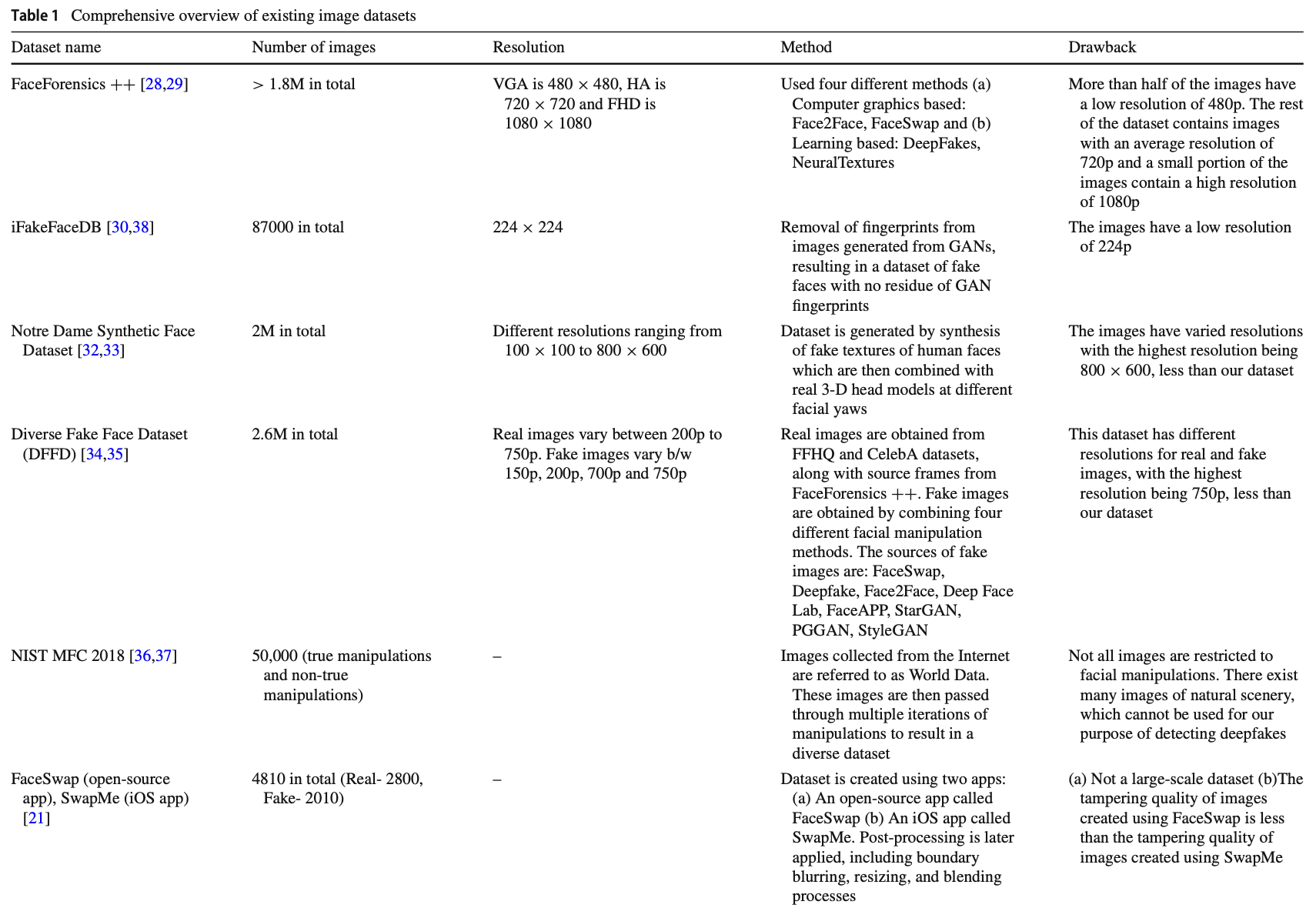
**4.4 DeepFake Image-high-quality (DFIM-HQ)**

DeepFake Image-high-quality (DFIM-HQ) has been created with the intention of making deepfakes dataset free from any kind of bias or constraints. Most of the datasets show bias towards particular gender,age and race. To overcome this, DFIM-HQ has been created created without any race,gender and age bias, where it includes pictures of people from young age to old age, all races and all genders. It also takes care of various scenarios such as image orientation changes, low quality images and illumination degradations. The dataset contains 70000 real images and 70000 fake images, which have been divided further as train,validation and test datasets. Train in each contains 52500 images, Validation contains 10500 images and Test contains 7000 images.

**4.5 Summary of datasets**

****

****

****

**5. System Requirements Specification**

**5.1 Introduction**

**5.1.1 Project Scope**

* Our project aims to develop a system that can detect deepfakes in videos with high accuracy, which is robust against image orientation changes, image quality and occlusion.
* The project scope includes features such as video analysis and classification of video.
* The target audience for the system is media companies, law enforcement agencies, and other organizations that need to detect deepfakes in videos.

**5.2 Product Perspective**

**5.2.1 Product Features**.

* The system will analyze video content to identify signs of manipulation, such as inconsistent facial expressions or lighting, and use machine learning algorithms to detect deepfakes with high accuracy.
* When a deepfake is detected, the system will trigger an alert and generate a report with details about the video and the detection results.

**5.2.2 Operating Environment**

* The deepfake detection system will require high-performance hardware, including a powerful CPU and GPU, and a large amount of memory and storage.
* The software requirements include operating systems such as Windows and Linux, and deep learning frameworks such as TensorFlow and PyTorch.

**5.2.3 General Constraints, Assumptions and Dependencies**

* The project depends on the availability of diverse and reliable data sources
* The project’s accuracy depends on the quality of the training data
* The project assumes that there will be a large number of deepfakes in the wild, and that the system will need to be updated regularly to keep up with new types of deepfakes.
* The project assumes that deepfake techniques used in the dataset are similar to those used in the real world
* The project assumes that the entire length of the video will either be real or deepfake
* The project may be constrained by limited resources, such as time and funding, and dependencies on third-party software libraries and frameworks.

**5.2.4 Risks**

* The system may produce false positives, which may lead to innocent users being falsely accused of creating or sharing deepfake
* The system may infringe on user privacy by analyzing and storing user data, leading to privacy concerns and potential legal issues
* The system may be vulnerable to adversarial attacks, where attackers may attempt to deceive the system by creating deepfakes that can evade detection
* Other risks include the potential for malicious actors to develop new types of deepfakes that the system cannot detect, and the risk of false confidence in the system's detection results.

**5.3Functional Requirements**

* The deepfake detection system must be able to analyze videos for signs of manipulation, such as inconsistent facial expressions or lighting, and use machine learning algorithms to detect deepfakes with high accuracy.
* Data Preprocessing: The system should be able to preprocess the dataset to extract relevant features from the images or videos to train the model
* Model Training: The system should be capable of training deep learning models to detect deepfakes
* Model Testing: The system should be able to test the model on a separate dataset to evaluate its performance
* Real-Time Detection: The system should be able to perform real-time deepfake detection on images and videos as they are uploaded
* Accuracy: The system should have a high accuracy rate in detecting deepfakes and minimizing false positives

**5.4 External Interface Requirements**

**5.4.1 User Interfaces**

The deepfake detection system will have a web-based user interface that allows users to upload videos for analysis and view detection results.

**5.4.2 Hardware Requirements**

The deepfake detection system will require high-performance hardware, including a powerful CPU and GPU, and a large amount of memory and storage.

**5.4.3 Software Requirements**

The software requirements include operating systems such as Windows and Linux, and deep learning frameworks such as TensorFlow,PyTorch,DlibNumpy,Pickle and OpenCV.

**5.4.4 Communication Interfaces**

The deepfake detection system will require communication interfaces, such as APIs or network connections, to integrate with other systems or applications.

**5.5 Non-Functional Requirements**

**5.5.1 Performance Requirements**

* The deepfake detection system must be able to analyze videos quickly and accurately, with a high detection rate and low false positive rate.
* The system must also be scalable, able to handle large volumes of video content and multiple users simultaneously.

**5.5.2 Safety Requirements**

* The deepfake detection system must not cause harm or endanger users in any way.
* It must not leak user data or use any user data without their consent

**5.5.3 Security Requirements**

The deepfake detection system must ensure the confidentiality, integrity, and availability of user data, and must be designed to prevent cyber attacks.

**5.6 Other Requirements**

The deepfake detection system must comply with legal and regulatory requirements, such as data privacy laws and intellectual property rights.

**6. System Design**

**6.1 Introduction**

This section provides an overview of the proposed system design for deepfake detection in videos, with a focus on facial visualization part .

**6.2 Current System**

Existing deepfake detection systems have limitations in detecting deepfakes that primarily focus on facial visualization. These techniques are not that effective in detecting sophisticated deepfakes. They very much rely on detecting artifacts, such as inconsistent lightning or blurriness that are introduced during the manipulation process.However, with the use of advanced machine learning algorithms, it is becoming increasingly difficult to detect these artifacts and deepfakes are becoming more realistic and sophisticated.

There is a need for a deepfake detection system that specifically focuses on the facial visualization part of the videos and uses advanced machine learning techniques to detect sophisticated deepfakes in real-time and at scale.The current system is also proven to fail when there is a change in orientation.The proposed system aims to address these limitations and provide a more effective and reliable solution for deepfake detection.

**6.3 Design Considerations**

**6.3.1 Design Goals**

* Higher accuracy even for the low quality images
* Real-time performance for deepfake videos
* Ability to detect sophisticated deepfakes
* Good model performance even when the the face orientation is bad

**6.3.2 Architecture Choices**

* Convolutional Neural Networks (CNNs): CNNs have been shown to be effective at detecting deepfakes in videos. One approach is to use a pre-trained CNN on image recognition tasks to extract features from individual frames of a video, and then use a recurrent neural network (RNN) to classify the video as real or fake based on the temporal sequence of these features.
* Two-stream CNNs: Two-stream CNNs use separate networks to analyze spatial and temporal information in videos. One stream processes individual frames, while the other analyzes motion between frames. These streams are then combined to classify the video as real or fake.
* Siamese Networks: Siamese networks are a type of neural network architecture that compares two inputs and determines whether they are similar or dissimilar. For deepfake detection, a Siamese network can be trained to compare pairs of frames from a video, and classify the video as real or fake based on the similarity of the frames.
* Grad-CAM (Gradient-weighted Class Activation Mapping): is a visualization technique used to understand which regions of an image a convolutional neural network (CNN) is focusing on when making a particular classification decision. It works by taking the gradient of the class score with respect to the feature maps of a CNN layer and then weighting these feature maps by their gradient values. This produces a heat map that highlights the regions of the input image that are most relevant to CNN's decision. Grad-CAM has been widely used in computer vision research and has applications in fields such as object detection, image segmentation, and visual question answering.
* Multi-Attentional Maps (MAM): It is a technique used to visualize the attention mechanisms in neural networks, particularly in natural language processing tasks such as language translation and sentiment analysis. MAM works by identifying the most relevant words or phrases in a given sentence or document, based on their relative importance to the task at hand. This is done by training a neural network to generate attention weights for each word in the input sequence, and then using these weights to create a heat map that highlights the words with the highest attention scores. MAM can be used to gain insights into how neural networks process and understand natural language, and can be used to improve the performance of natural language processing systems.

**6.3.3 Constraints, Assumptions and Dependencies**

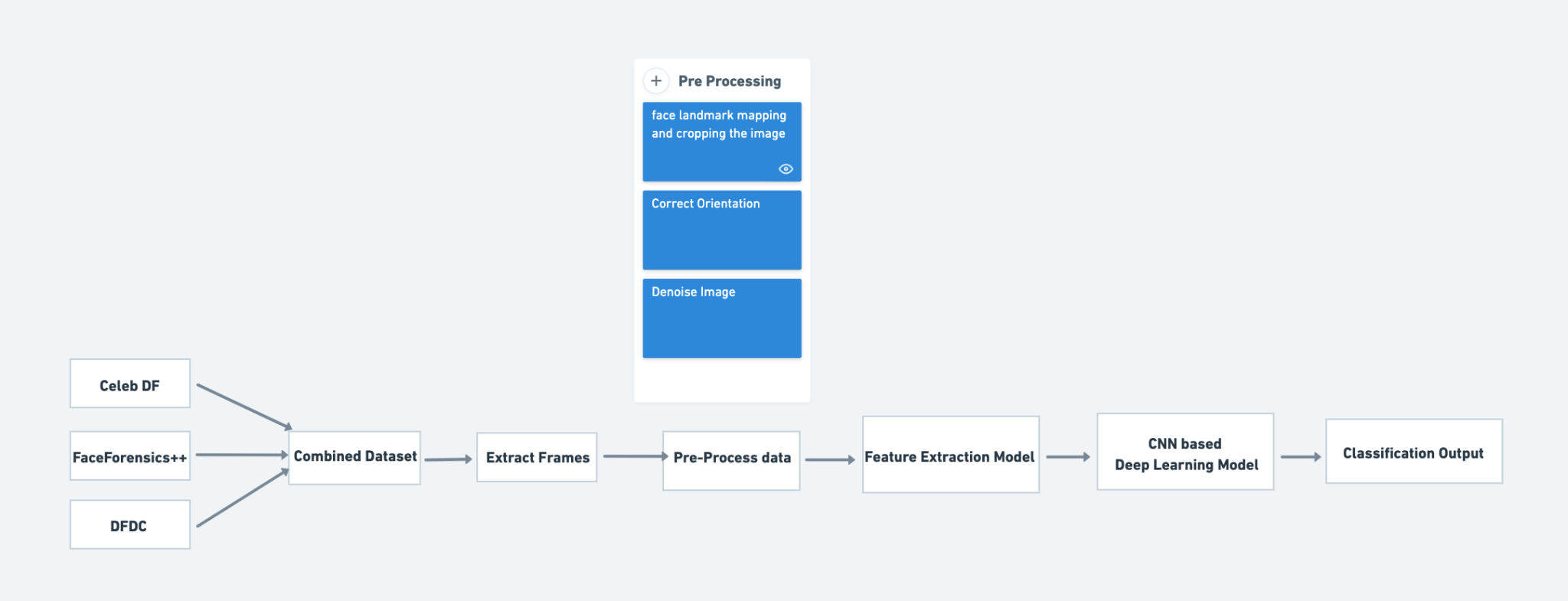
* The model assumes a uniform video data format.
* The model assumes that the entire length of the video will either be real or deepfake.
* The model’s accuracy depends on the quality of training data.
* The model cannot have a fixed size for the feature extraction kernel.
* The project depends on the availability of diverse and reliable data sources
* The project’s accuracy depends on the quality of the training data
* The project assumes that there will be a large number of deepfakes in the wild, and that the system will need to be updated regularly to keep up with new types of deepfakes.
* The project assumes that deepfake techniques used in the dataset are similar to those used in the real world
* The project assumes that the entire length of the video will either be real or deepfake
* The project may be constrained by limited resources, such as time and funding, and dependencies on third-party software libraries and frameworks.

**6.4 High Level System Design**

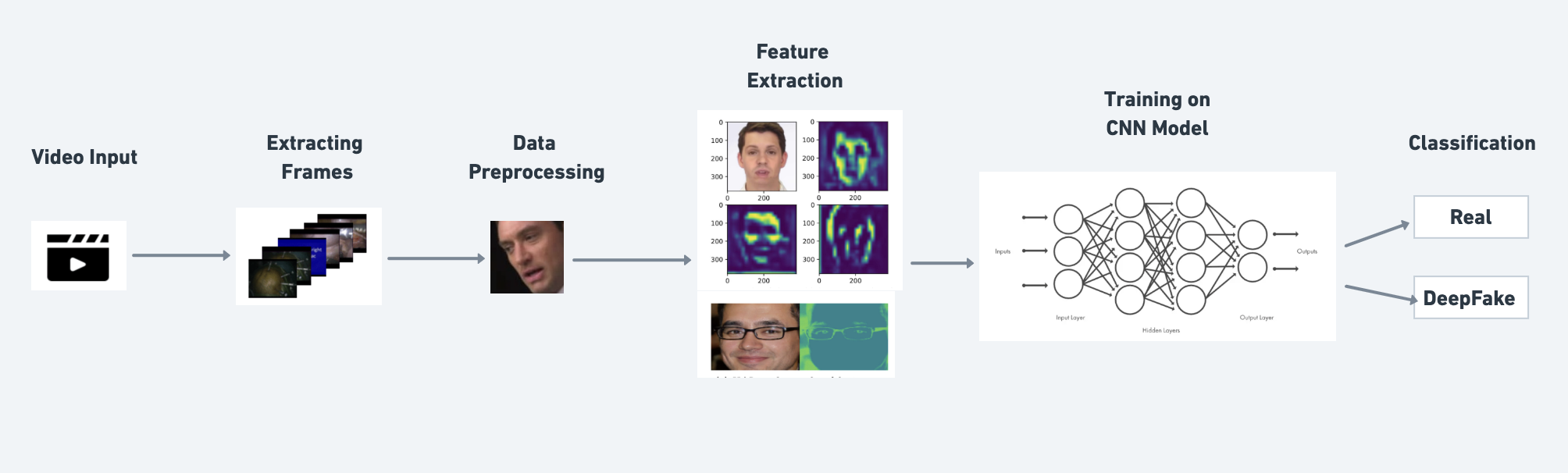
**6.4.1 Steps**

* Obtain datasets from references research
* Merge all the datasets,extract frames from videos and pre process it
* Pass the processed data through feature extraction model
* Pass the above obtained data to deep learning model
* Predict the output

**6.4.2 Block Diagram**

****

**6.4.3 High Level Design Diagram**

****

**6.5 Design Details**

**6.5.1 Novelty**

* Effective on low quality images also
* Robust against bad orientation of face
* Works even when occlusion occurs

**6.5.2 Innovativeness**

The project can be considered innovative in its approach to detecting deepfakes in real-time or in its use of novel deep learning architectures.

**6.5.3 Interoperability**

The deepfake detection project can work with different forms of video files, such as mp4,mov,mkv,etc,. It will also be compatible with different platforms, such as mobile devices or cloud-based services. Ensuring interoperability may involve designing an appropriate data schema, creating adapters for different media formats, or using standardized APIs.

**6.5.4 Performance**

The performance of our model will be the same on low quality images or bad face orientations. The model will trained and optimized properly to get higher accuracies and faster detection speeds

**6.5.5 Security**

The deepfake detection project incorporates advanced security measures, such as using secure communication protocols and encryption, to prevent data leakage and protect against attacks. It also uses adversarial training and other techniques to make the model more robust to attacks and manipulations.

**6.5.6 Reliability**

The deepfake detection project is highly reliable, achieving consistent levels of accuracy across different types of deepfakes and media file formats. It is also designed to be easily scalable and adaptable to different contexts, ensuring reliability in production environments.

**6.5.7 Maintainability**

The deepfake detection project is designed with maintainability in mind, using modularized code and automated testing to make it easy to manage code changes and updates. It also incorporates version control tools and other software development best practices to ensure long-term maintainability.

**6.5.8 Portability**

The deepfake detection project is designed to be portable across different hardware and software environments.

**6.5.9 Legacy to Modernization**

The deepfake detection project is designed to transition from older, rule-based methods of detecting manipulated media to newer, machine learning-based methods. It uses algorithms to automatically generate labeled data and collects a large dataset of manipulated and unmanipulated media files to improve detection accuracy.

**6.5.10 Reusability**

The deepfake detection project is designed to be reusable across different applications and contexts, using standard APIs to make it easy to integrate the system with other software tools and platforms. It also incorporates custom adapters to handle different media formats and ensure compatibility with different applications.

**6.5.11 Application Compatibility**

The deepfake detection project is designed to be compatible with different types of applications, such as social media platforms and video editing software. It incorporates custom adapters to handle different media formats.

**7. IMPLEMENTATION AND PSEUDOCODE**

The below code shows the preprocessing part of the code, in which the data from the dataset is extracted and stored in different directories. The videos are converted to frames and they are dynamically selected and stored based on the number of frames in the video. Later the frames are cropped to get only the face part and they are stored in a separate directory

**7.1 Code**

from zipfile import ZipFile

import dlib

import os

import cv2

local\_zip = "/content/celeb-df-v2.zip"

zip\_ref = *ZipFile*(*local\_zip*,'r')

zip\_ref.*extractall*("/content/drive/MyDrive/celebdf")

zip\_ref.*close*()

*def* no\_of\_frames\_to\_skip(frames):

frames = *int*(*frames* /60)

res = 5

if frames <= 1:

return res

return res \* frames

*def* make\_frames\_from\_video(video\_path, video\_id):

input\_file = video\_path

# *output\_folder = '/content/drive/MyDrive/celebdf/synthetic-frames'*

# *output\_folder = '/content/drive/MyDrive/celebdf/original-frames'*

output\_folder = '/content/drive/MyDrive/celebdf/new-original-frames'

if not os.path.*exists*(*output\_folder*):

os.*makedirs*(*output\_folder*)

try:

video = cv2.*VideoCapture*(*input\_file*)

fps = video.*get*(*cv2*.*CAP\_PROP\_FPS*)

frame\_count = *int*(*video*.*get*(*cv2*.*CAP\_PROP\_FRAME\_COUNT*))

frame\_skips = *no\_of\_frames\_to\_skip*(*frame\_count*)

frame\_num = 0

saved\_frames = 0

while True:

ret, frame = video.*read*()

if not ret:

break

if frame\_num % frame\_skips == 0:

output\_file = os.path.*join*(*output\_folder*, *f*'*{video\_id}*@*{saved\_frames:06d}*.jpg')

cv2.*imwrite*(*output\_file*, *frame*)

saved\_frames += 1

frame\_num += 1

video.*release*()

except Exception as e:

*print*("error: "+*e*)

# *input\_folder = '/content/drive/MyDrive/celebdf/Celeb-synthesis'*

input\_folder = '/content/drive/MyDrive/celebdf/Celeb-real'

for filename in os.*listdir*(*input\_folder*):

*make\_frames\_from\_video*(*input\_folder*+"/"+*filename*,*filename*)

*def* crop\_images():

detector = dlib.*get\_frontal\_face\_detector*()

# *input\_folder = '/content/drive/MyDrive/celebdf/synthetic-frames'*

# *output\_folder = '/content/drive/MyDrive/celebdf/synthetic-cropped-images'*

input\_folder = '/content/drive/MyDrive/celebdf/new-original-frames'

output\_folder = '/content/drive/MyDrive/celebdf/new-original-cropped-images'

if not os.path.*exists*(*output\_folder*):

os.*makedirs*(*output\_folder*)

for filename in os.*listdir*(*input\_folder*):

try:

image = cv2.*imread*(*os*.*path*.*join*(*input\_folder*, *filename*))

gray = cv2.*cvtColor*(*image*, *cv2*.*COLOR\_BGR2GRAY*)

faces = *detector*(*gray*)

for face in faces:

x1, y1, x2, y2 = face.*left*(), face.*top*(), face.*right*(), face.*bottom*()

face\_image = image[y1:y2, x1:x2]

cv2.*imwrite*(*os*.*path*.*join*(*output\_folder*, *filename*), *face\_image*)

except Exception as e:

*print*(*e*)

**7.2 Output**

Original image:

****

Cropped image:



**8. CONCLUSION OF CAPSTONE PROJECT PHASE-1**

Phase-1 of Capstone Project named “DeepFake Detection For Videos” ensures the completion of below mentioned milestones:

* Completed research survey.
* We have finalised gaps to be worked upon.
* Collecting datasets.
* Basic preprocessing of the data.

**9. PLAN OF WORK FOR CAPSTONE PROJECT PHASE-2**

In the phase-2 of the Capstone Project we are planning to work on the below mentioned points:

* Do more preprocessing on the datasets and combine the datasets.
* Test existing architectures by implementing them on the new combined dataset.
* Implementation of our new model.
* Testing, Improving the accuracy and Parameters fine tuning.
* Work on a research paper for publishing it in journals or conferences.

**REFERENCE/ BIBLIOGRAPHY**

[1] Mathews, Sherin, et al. "An explainable deepfake detection framework on a novel unconstrained dataset." Complex & Intelligent Systems (2023): 1-13.

[2] Patil, Kundan, et al. "Deepfake Detection using Biological Features: A Survey." arXiv preprint arXiv:2301.05819 (2023).

[3] Zhao, Hanqing, et al. "Multi-attentional deepfake detection." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021.

[4] Wodajo, Deressa, and Solomon Atnafu. "Deepfake video detection using convolutional vision transformer." arXiv preprint arXiv:2102.11126 (2021).

[5] Zhang, Li, et al. "Unsupervised learning-based framework for deepfake video detection." IEEE Transactions on Multimedia (2022).

[6] Ikram, Sumaiya Thaseen, Shourya Chambial, and Dhruv Sood. "A Performance Enhancement of Deepfake Video Detection through the use of a Hybrid CNN Deep Learning Model." International journal of electrical and computer engineering systems 14.2 (2023): 169-178.

[7] Li, Xiaodan, et al. "Sharp multiple instance learning for deepfake video detection." Proceedings of the 28th ACM international conference on multimedia. 2020.

[8] Das, S., Seferbekov, S., Datta, A., Islam, M. and Amin, M., 2021. Towards solving the deepfake problem: An analysis on improving deepfake detection using dynamic face augmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 3776-3785).

[9] Guarnera, L., Giudice, O. and Battiato, S., 2020. Fighting deepfake by exposing the convolutional traces on images. IEEE Access, 8, pp.165085-165098.

[10] Tan, Z., Yang, Z., Miao, C. and Guo, G., 2022. Transformer-based feature compensation and aggregation for deepfake detection. IEEE Signal Processing Letters, 29, pp.2183-2187.

[11] Coccomini, D.A., Messina, N., Gennaro, C. and Falchi, F., 2022, May. Combining efficientnet and vision transformers for video deepfake detection. In Image Analysis and Processing–ICIAP 2022: 21st International Conference, Lecce, Italy, May 23–27, 2022, Proceedings, Part III (pp. 219-229). Cham: Springer International Publishing.

[12] V. -N. Tran, S. -G. Kwon, S. -H. Lee, H. -S. Le and K. -R. Kwon, "Generalization of Forgery Detection With Meta Deepfake Detection Model," in IEEE Access, vol. 11, pp. 535-546, 2023, doi: 10.1109/ACCESS.2022.3232290.

[13] W. Lu et al., "Detection of Deepfake Videos Using Long-Distance Attention," in IEEE Transactions on Neural Networks and Learning Systems, doi: 10.1109/TNNLS.2022.3233063.

[14] Y. Nirkin, L. Wolf, Y. Keller and T. Hassner, "DeepFake Detection Based on Discrepancies Between Faces and Their Context," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 10, pp. 6111-6121, 1 Oct. 2022, doi: 10.1109/TPAMI.2021.3093446.

[15] H. A. Khalil and S. A. Maged, "Deepfakes Creation and Detection Using Deep Learning," 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Cairo, Egypt, 2021, pp. 1-4, doi: 10.1109/MIUCC52538.2021.9447642.

[16] H. Chen, Y. Lin, B. Li and S. Tan, "Learning Features of Intra-consistency and Inter-diversity: Keys towards Generalizable Deepfake Detection," in IEEE Transactions on Circuits and Systems for Video Technology, 2022, doi: 10.1109/TCSVT.2022.3209336.

[17] C. R. Gerstner and H. Farid, "Detecting Real-Time Deep-Fake Videos Using Active Illumination," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 2022, pp. 53-60, doi: 10.1109/CVPRW56347.2022.00015.

[18] J. John and B. V. Sherif, "Comparative Analysis on Different DeepFake Detection Methods and Semi Supervised GAN Architecture for DeepFake Detection," 2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Dharan, Nepal, 2022, pp. 516-521, doi: 10.1109/I-SMAC55078.2022.9987265.

[19] A. Chintha et al., "Recurrent Convolutional Structures for Audio Spoof and Video Deepfake Detection," in IEEE Journal of Selected Topics in Signal Processing, vol. 14, no. 5, pp. 1024-1037, Aug. 2020, doi: 10.1109/JSTSP.2020.2999185.

[20] A. Rahman et al., "Short And Low Resolution Deepfake Video Detection Using CNN," 2022 IEEE 10th Region 10 Humanitarian Technology Conference (R10-HTC), Hyderabad, India, 2022, pp. 259-264, doi: 10.1109/R10-HTC54060.2022.9929719.