TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

LALITPUR ENGINEERING COLLEGE
KHOLKA POKHARI, LALITPUR

A PROPOSAL OF MAJOR PROJECT

DefaceLab: DeepFake Detection using Deep Learning

SUBMITTED BY

ABHISHEK NEUPANE [LEC076BCT002]
RABINDRA ADHIKARI [LEC076BCT025]
SANJISH MAHARJAN [LEC076BCT032]
SUSHIL KAFLE [LEC076BCT045]

SUBMITTED TO

DEPARTMENT OF COMPUTER ENGINERING

2080-02-25

DefaceLab: DeepFake Detection using Deep Learning

Submitted by
ABHISHEK NEUPANE [LEC076BCT002]
RABINDRA ADHIKARI [LEC076BCT025]
SANJISH MAHARJAN [LEC076BCT032]
SUSHIL KAFLE [LEC076BCT045]

Project Coodinator
Er. Bishika Subedi

A project submitted in partial fulfillment of the requirements for the degree of
Bachelor of Computer Engineering

DEPARTMENT OF COMPUTER ENGINEERING
Lalitpur Engineering College
Tribhuvan University

2080-02-25

ABSTRACT

Deepfakes are realistic-looking fake media generated by deep-learning algorithms that iterate through large datasets until they have learned how to solve the given problem (i.e., swap faces or objects in video and digital content). The massive generation of such content and modification technologies is rapidly affecting the quality of public discourse and the safeguarding of human rights. Deepfakes are being widely used as a malicious source of misinformation in court that seek to sway a court's decision. Because digital evidence is critical to the outcome of many legal cases, detecting deepfake media is extremely important and in high demand in digital forensics. As such, it is important to identify and build a classifier that can accurately distinguish between authentic and disguised media, especially in facial-recognition systems as it can be used in identity protection too. In this work, we compare the most common, state-of-the-art face-detection classifiers such as Custom CNN, VGGface2, and DenseNet-121 using an augmented real and fake face-detection dataset. Data augmentation is used to boost performance and reduce computational resources. Our preliminary results indicate that VGG19 has the best performance and highest accuracy of 95% when compared with other analyzed models.

Keywords: deepfake detection; digital forensics; media forensics; deep learning; VGGface2; face-image manipulation

Contents

List of Figures

Abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DL	Deep Learning
FPS	Frames Per Second
GAN	Generative Adversarial Network
ML	Machine Learning
RNN	Recurrent Neural Network

1 INTRODUCTION

Deepfake technology has revolutionized the world of digital media manipulation. By combining artificial intelligence and image/video processing, deepfakes have garnered widespread attention. Deepfakes involve the creation of realistic media portraying individuals in situations they never experienced or saying things they never said. As this technology becomes more sophisticated, concerns arise regarding its impact on politics, entertainment, and personal privacy. This report provides an overview of deepfakes, including their underlying processes, societal implications, ethical challenges and the way of detection. In navigating this landscape, it is crucial to find a balance between innovation and responsible use in our increasingly digitized society.

In the last few years, cybercrime, which accounts for a 67% increase in the incidents of security breaches, has been one of the most challenging problems that national security systems have had to deal with worldwide.[1] Deepfakes, at present time, are being widely used to swap faces or objects in video and digital content. This artificial intelligence-synthesized content can have a significant impact on the determination of legitimacy due to its wide variety of applications and formats that deepfakes present online (i.e., audio, image and video). Considering the quickness, ease of use, and impacts of social media, persuasive deepfakes can rapidly influence millions of people, destroy the lives of its victims and have a negative impact on society in general [1]. Deepfake technology has been driven by various motivations, including individual attacks, political manipulation, and the spread of false information. Its impact extends beyond personal attacks to manipulating satellite images and using stock images for identity protection. Cyber attackers continuously adapt their strategies, making it challenging to identify deepfake media and stay ahead of evolving threats.

The societal implications of deepfake technology are profound. Misinformation and disinformation fueled by deepfakes erode public trust, damage reputations, and violate privacy at personal and professional levels. Deepfakes can also disrupt democratic processes and contribute to societal polarization. Addressing the legal and ethical concerns surrounding deepfakes requires technological advancements, policy development, media literacy, and careful consideration of privacy rights and the manipulation of visual evidence. To tackle these implications, it is essential to advance deepfake detection methods, bolster cybersecurity measures, promote media literacy for individuals to discern manipulated content, and establish clear legal frameworks governing the responsible use of deepfake technology. By taking a comprehensive approach, we can effectively navigate the ethical challenges and societal impacts posed by deepfakes.

The deepfake technology holds importance in several areas. It offers creative expression and entertainment possibilities, enhances research and development in fields like computer vision, and aids forensic analysis in legal investigations. Deepfakes also emphasize the need for media literacy and critical thinking skills, promoting education and awareness. Ethical considerations and policy development are crucial in addressing the responsible use of deepfakes and protecting individuals' rights. Understanding the significance of deepfake technology enables us to navigate its implications effectively and harness its potential while mitigating potential harm.

We cannot dispute the influence deep fakes will have in the next years given all the benefits and cons that have been presented. Therefore, keeping an eye on deepfake content is crucial. This paper will provide an overview of the fundamental organizational structure of our project on how deepfake detections can be done.

1.1 Background

At present context of time, the rapid advancements in mobile camera technology and the widespread use of social media platforms have made it easier than ever to create and share digital pictures. Deep learning has played a crucial role in developing technologies that were previously unimaginable. One notable example is modern generative models, which can produce highly realistic images, speech, music, and video. These models have been applied in various fields, such as enhancing accessibility through text-to-speech technology and generating training data for medical imaging.

There will always be drawbacks to any technological breakthrough. Since deepfakes are still relatively new and expanding quickly, their excessive use as a result of rising human interest has resulted in misuse of this technology. It is simple for widespread false information to proliferate among the populace when there is no controlling element and a weak mechanism in place to identify deep fakes.Since their initial emergence in late 2017, a variety of open-source deep fake generation techniques and tools have appeared, resulting in an increase in the amount of synthetic media clips. Others may be destructive to people and society, even though many are probably intended to be amusing. Due to the accessibility of editing tools and the strong demand for topic expertise, false digital contents have been growing in number and in realism up until recently.

Deep fakes are now widely disseminated on social media platforms, which encourages spamming and the spread of false information. Just picture a deep fake image of Donald Trump getting arrested which was trending on twitter or a deep fake of a well-known celebrity assaulting their supporters. These types of misinformation can brainwash the audience and are awful and endanger and mislead the general public.

Deep fake detection plays a crucial part in overcoming such a circumstance. Therefore, we provide a novel deep learning-based method that can successfully separate artificial intelligence-generated fake contents from authentic digital materials. In order to identify deep fakes and stop them from spreading across the internet, it is crucial to develop technology that can detect deepfakes.

1.2 Problem Statement

With the help of visual effects, convincing modifications of digital photographs and videos have been proven for many years. However, new developments in deep learning have dramatically increased the realism of fake content and made it more widely available. These purportedly artificial intelligence-generated works of media are also known as ”deepfakes”. It is easy to create deep fakes utilizing artificial intelligence techniques. However, it is extremely difficult to identify these Deep Fakes. In the past, there have been numerous instances of deep fakes being used to effectively incite political unrest, stage terrorist attacks, blackmail individuals, etc. Therefore, it becomes crucial to identify these deep fakes and stop their spread through social media. Therefore, with the growing curiosity we have taken a step forward in detecting the deep fakes using vggface2 based artificial neural network.

1.3 Objectives

- Our project can help in reduction in spread of false informations, that might mislead the people on the internet.
- Our project will distinguish and classify the video as deepfake or pristine.

1.4 Scope

At present time there are numerous tools available for creating false videos in the current deepfake technology landscape, but there are few trustworthy tools available for spotting them. The idea creation of a deepfake detection software to solve this discrepancy and stop the widespread dissemination of deepfakes is what our project is based upon. Users will be able to post images through our platform and segregate them as authentic or deepfake. This project can also be developed to include the development of a plugin for browsers that will automatically detect deep fakes. Notably, our idea can me implemented on different social sites as well as in various various governmental organizations. A synopsis of the program with the size of the input, bounds on the input, input validation, input dependency, the i/o state diagram, and the major inputs and outputs are explained in this report.

2 LITERATURE REVIEW

Generative adversarial networks (GANs) are a type of deep neural network commonly used for creating deepfakes. GANs have the advantage of being able to learn from training data and generate new data with similar features and characteristics. The architecture includes an encoder and decoder, where the encoder learns from a dataset to create fake data, and the decoder learns to differentiate between real and fake data[]. However, GANs require a substantial amount of data to generate realistic-looking faces.

FakeApp is a widely used method for creating deepfakes, allowing face swapping in videos using an autoencoder-decoder structure. It can generate highly realistic fake videos that are difficult to distinguish from real ones. VGGFace, another popular deepfake technique, utilizes a generative adversarial network (GAN) and improves the architecture with additional layers for adversarial and perceptual losses. These enhancements capture facial features such as eye movements, resulting in more believable and realistic fake images.

CycleGAN [3] is a deepfake technique that extracts the characteristics of one image and produces another image with the same characteristics via the GAN architecture. This method applies cycle loss function that enables them to learn the latent features. Dissimilar from FakeApp, CycleGAN is unsupervised method that can perform image-to-image conversion without using paired examples.

Recurrent Neural Network [4] (RNN) for deepfake detection used the approach of using RNN for sequential processing of the frames along with ImageNet pre-trained model. Their process used the HOHO dataset consisting of just 600 videos. This dataset consists small number of videos and same type of videos, which may not perform very well on the real time data. We will be training out model on large number of Realtime data.

MegaFace dataset [6] was released in 2016 to evaluate face recognition methods with up to a million distractors in the gallery image set. It contains 4.7 million images of 672, 057 identities as the training set. However, an average of only 7 images per identity makes it restricted in its per identity face variation. In order to study the effect of pose and age variations in recognising faces, the MegaFace challenge [6] uses the subsets of FaceScrub [3] containing 4, 000 images from 80 identities and FG-NET [4] containing 975 images from 82 identities for evaluation.

The VGGFace2 dataset contains 3.31 million images from 9131 celebrities spanning a wide range of ethnicities.The dataset is approximately gender-balanced, with 59.3%

males, varying between 80 and 843 images for each identity, with 362.6 images on average. It includes human verified bounding boxes around faces, and five fiducial keypoints predicted by the model of [2].The VGGFace2 provides annotation to enable evaluation on two scenarios: face matching across different poses, and face matching across different ages.

A convolutional neural network (CNN) is the most commonly used deep neural network model. CNN, like neural networks, has an input and output layer, as well as one or more hidden layers. In CNN [4], the hidden layers first read the inputs from the first layer and then apply a convolution mathematical operation on the input values. Here, convolution indicates a matrix multiplication or other dot product. After applying matrix multiplication, CNN uses the nonlinearity activation function such as Rectified Linear Unit (RELU) followed by additional convolutions such as pooling layers. The main goal of pooling layers is to reduce the dimensionality of the data by computing the outputs utilizing functions such as maximum pooling or average pooling.

Microsoft released the large Ms-Celeb-1M dataset [7] in 2016 with 10 million images from 100k celebrities for training and testing. This is a very useful dataset, and we employ it for pre-training in this project. However,[] it has two limitations: (i) while it has the largest number of training images, the intra-identity variation is somewhat restricted due to an average of 81 images per person; (ii) images in the training set were directly retrieved from a search engine without manual filtering, and consequently there is label noise.

2.1 Existing Systems

2.1.1 Deepware

Deepware.ai is an innovative company at the forefront of deepfake detection technology. They specialize in developing advanced AI-driven solutions to combat the spread of manipulated media content. With a team of expert researchers and engineers, Deepware.ai leverages state-of-the-art machine learning algorithms and deep neural networks to accurately identify deepfakes. Their cutting-edge technology, combined with a user-friendly approach, empowers individuals and organizations to protect themselves from the potentially harmful consequences of deepfakes. Deepware.ai's commitment to continuous improvement and staying ahead of evolving deepfake techniques positions them as a trusted leader in the field, offering reliable and scalable solutions that contribute to a safer digital landscape.

Figure 1: Deepware

2.1.2 DuckDuckGoose

DuckDuckGoose offers an open-source browser extension that keeps tabs on all websites you visit and alert you once manipulated media is detected. Users should also appreciate the transparency of the DeepFake detector, as DuckDuckGoose provides detailed explanations for why a video was flagged to give you some insight on what to look for in a DeepFake. The team behind the tool has been dedicated to sharing their research findings and encouraging participants from the community to contribute to building a more reliable model with higher accuracy.

Figure 2: DuckDuckGoose

2.2 Proposed Systems

Our Deepfake Detection System is an online tool designed to help people identify and deal with deepfake content. It focuses on providing strong detection capabilities for deepfakes. Users can use the system to analyze and detect potential deepfakes by submitting images. The system uses advanced algorithms and machine learning techniques to accurately identify manipulated or fake media. This helps users recognize and address the risks that come with deepfakes, such as spreading false information, committing fraud, or violating privacy. The deepfake detection system aims to empower users by giving them the tools they need to protect themselves and others from the harmful effects of encountering deepfakes. With the help of cutting-edge algorithms, the system assists users in detecting and raising awareness about deepfakes, making the digital world a safer and more informed place.

3 FEASIBILITY STUDY

3.1 Economic feasibility

This is a low-budget project with no development costs. The total expenditure of the project is just computational power. The dataset and computational power required for the project are easily available. The computational power is easily provided by google collab. So, the project is economically feasible. The system will be simple to comprehend and use. As a result, there will be no need of trained personnel to use the system. This system will have the capacity to expand by adding more components.

3.2 Operational feasibility

The project is operationally feasible since after the completion of the project, it can be operated as intended by the user to solve the problems for what it has been developed.

3.3 Technical feasibility

The purpose of technical feasibility is to establish whether the project is possible in terms of software, hardware, manpower, and knowledge to complete. It will take into account determining resources in support of the suggested scheme. The system is platform independent because it is written in Python. Advanced machine learning libraries are available and the technology is cutting-edge. As a result, the system is technically possible.

4 METHODOLOGY

4.1 Software Development Life Cycle

Agile method of Software Development uses iterative approach. Agile method cycles among Planning, Requirement Analysis, Designing, Development and Testing stages. These cycle is called sprints. Each sprints are considered as a miniature project on itself. Using this method allowed us to update various parts of project at any point of project development. In this model an iterative approach was taken where working software was delivered after each iteration some new features is added to main system. It works in incremental and iterative approach. Agile model mainly focuses on customer collaborations, on individuals and iterations and welcomes changes at anytime in SDLC process. We prefer to use agile model in this system as it helps in developing realistic systems and promotes teamwork during software development. Also system is easy to manage and it can accommodate new changes at any stages of software development phase.

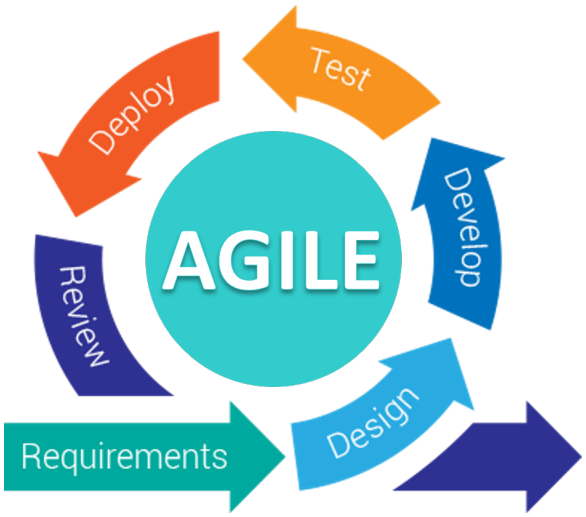

Figure 3: Agile Model

4.2 System Development Tools

Our static Deepfake detection System requires Python, Tensorflow, OpenCV, Machine Learning which are listed below:

1. Python
2. Pytorch
3. NumPy
4. OpenCV
5. Tensorflow

4.3 Functional Requirement

The functional requirements of the system are:

1. Detecting the Faces from Images and Videos.
2. Testing for realism of image.

4.4 Non Functional Requirement

These requirements are not needed by the system but are essential for the better performance of software. The points below focus on the non-functional requirement of the system.

- Reliability
- Usability
- Security
- Portability
- Speed and responsiveness
- Performance

4.5 Recurrent Neural Network

RNN, which stands for Recurrent Neural Network, is a type of artificial neural network commonly used for sequential data processing tasks. Unlike feedforward neural networks, which process data in a single forward pass, RNNs have a feedback connection that allows information to be fed back into the network. This enables RNNs to maintain an internal memory and process sequences of variable length.

The key feature of RNNs is their ability to capture temporal dependencies and learn from past information. Each input in a sequence is processed along with the internal state of the network, which is updated at each time step. This allows the network to incorporate context and previous information while making predictions or generating output.

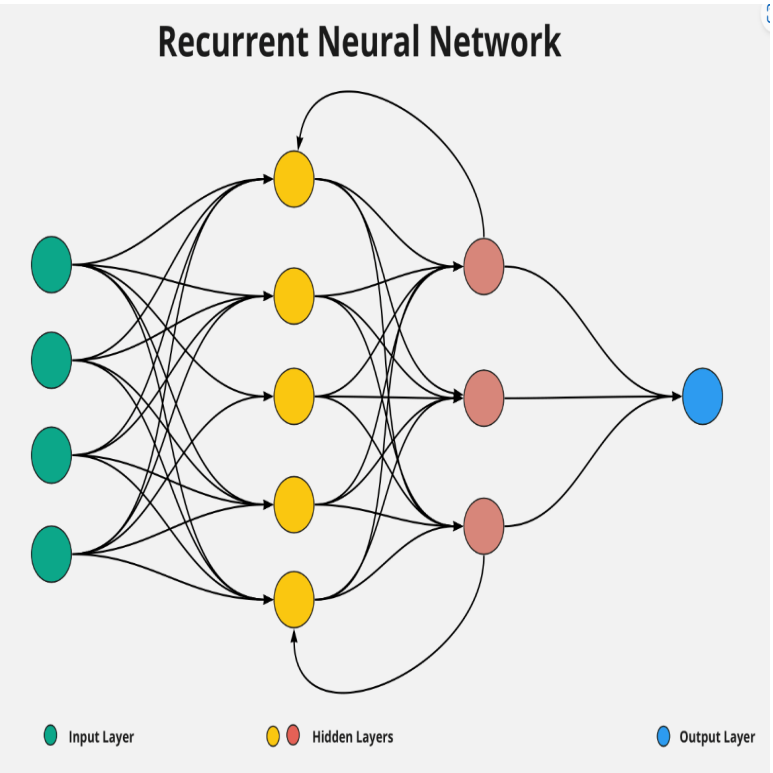

Figure 4: Recurrent Neural Network

4.6 Vggface2

Due to its broad scope and comprehensive coverage, VGGFace2 represents a significant leap in the field of facial recognition. It responds to the demand for a large dataset that can be used to train deep convolutional neural networks (CNNs) to accurately identify faces across a range of identities and demographics. VGGFace2 delivers a robust and wide set of facial data for training and assessing face recognition models, with over 3.3 million photos of more than 9,000 different people.

Each image in the collection has accurate and consistent information because of the rigorous annotation and labeling. The bounding box coordinates, which identify the face's placement inside the picture, are included in the annotations. This enables practitioners and academics to train and test face recognition algorithms that are purely concerned with the facial region.

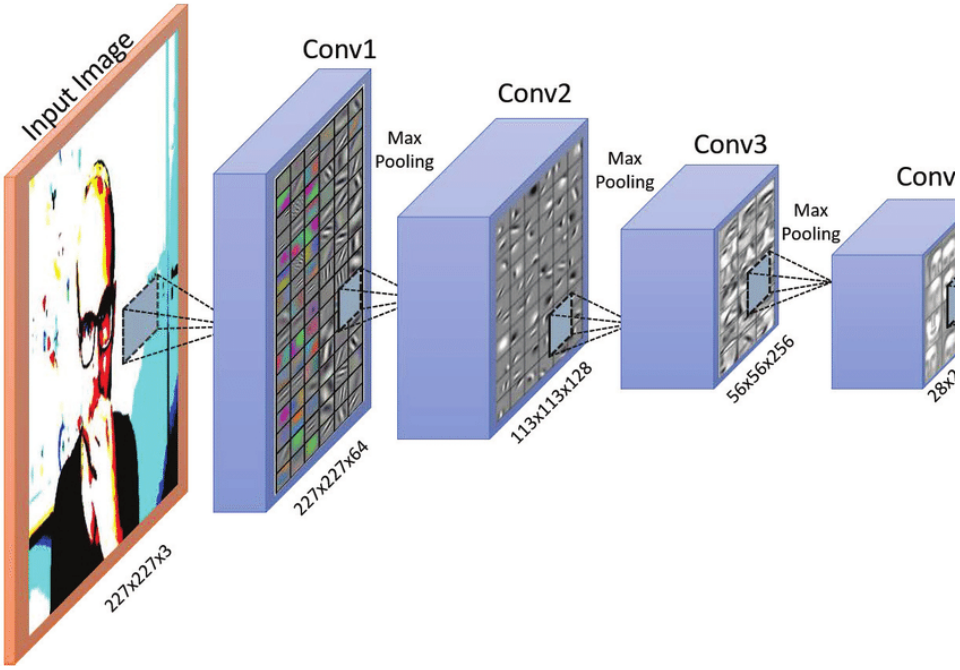

Figure 5: Vggface2 Architecture

4.7 MTCNN

Multi-task Cascaded Convolutional Networks, sometimes known as MTCNN, is a well-liked face identification technique that is frequently applied in computer vision applications. It is particularly made to locate and locate faces accurately in photos. The MTCNN performs face identification, alignment, and landmark localization using three cascaded networks.

The "Proposal Network" (P-Net), the first stage of MTCNN, creates a set of candidate bounding boxes that could include faces. The second step, known as the "Refine Network" (R-Net), which use regression and classification to increase the precision of the bounding box predictions, refines these candidate boxes. The third step, referred to as the "Output Network" (O-Net), completes additional refining and localisation of face landmarks.

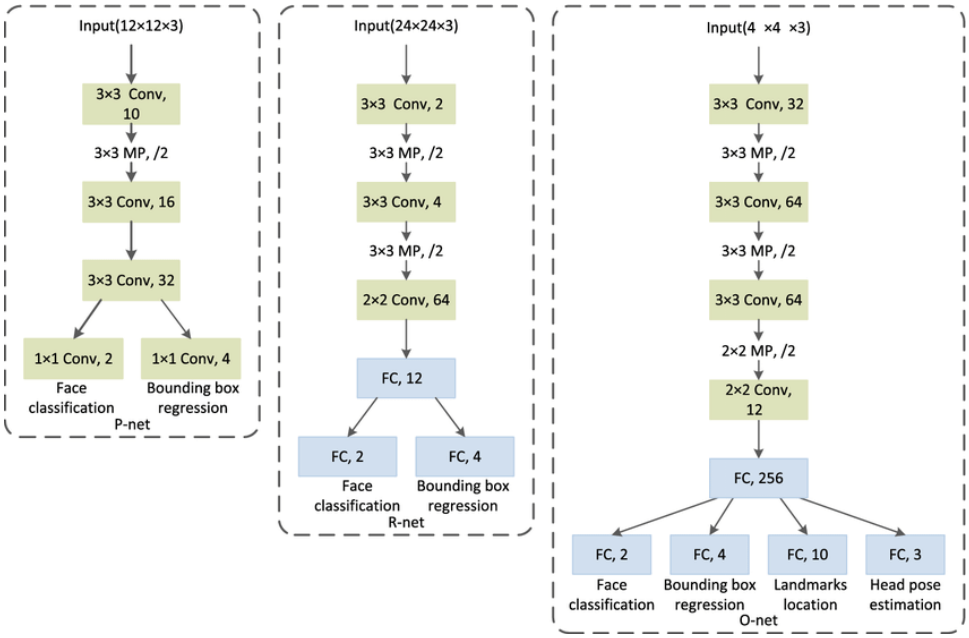

Figure 6: MTCNN Architecture

5 BLOCK DIAGRAMS

5.1 System Architecture

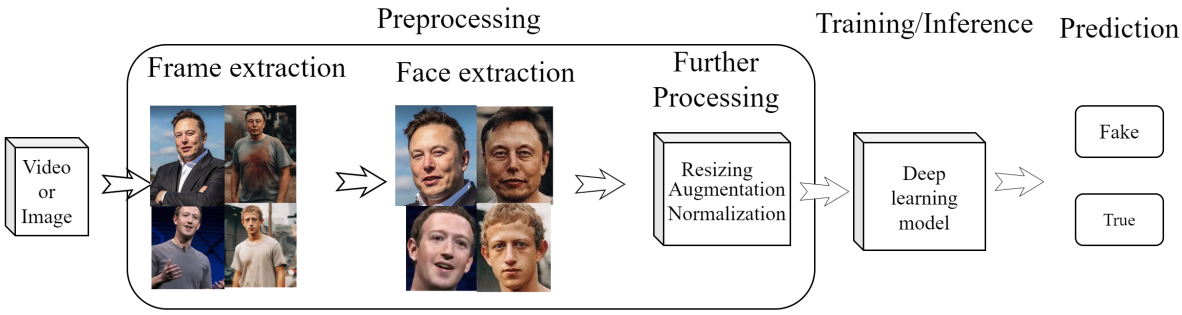

Figure 7: System Architecture

The system architecture of our project involves multiple steps. Initially, frames are extracted from the input video or obtained directly from an image. These frames then undergo face extraction, where faces are identified and cropped using face detection algorithms. The extracted faces are resized to a standardized size and undergo normalization to ensure consistent pixel values. The preprocessed face images are then fed into a deep learning model for classification. The model analyzes the features and patterns in the images to determine whether they are real or fake. Finally, the system produces the output, indicating the authenticity of the input video or image as either real or fake.

5.2 Use Case Diagram

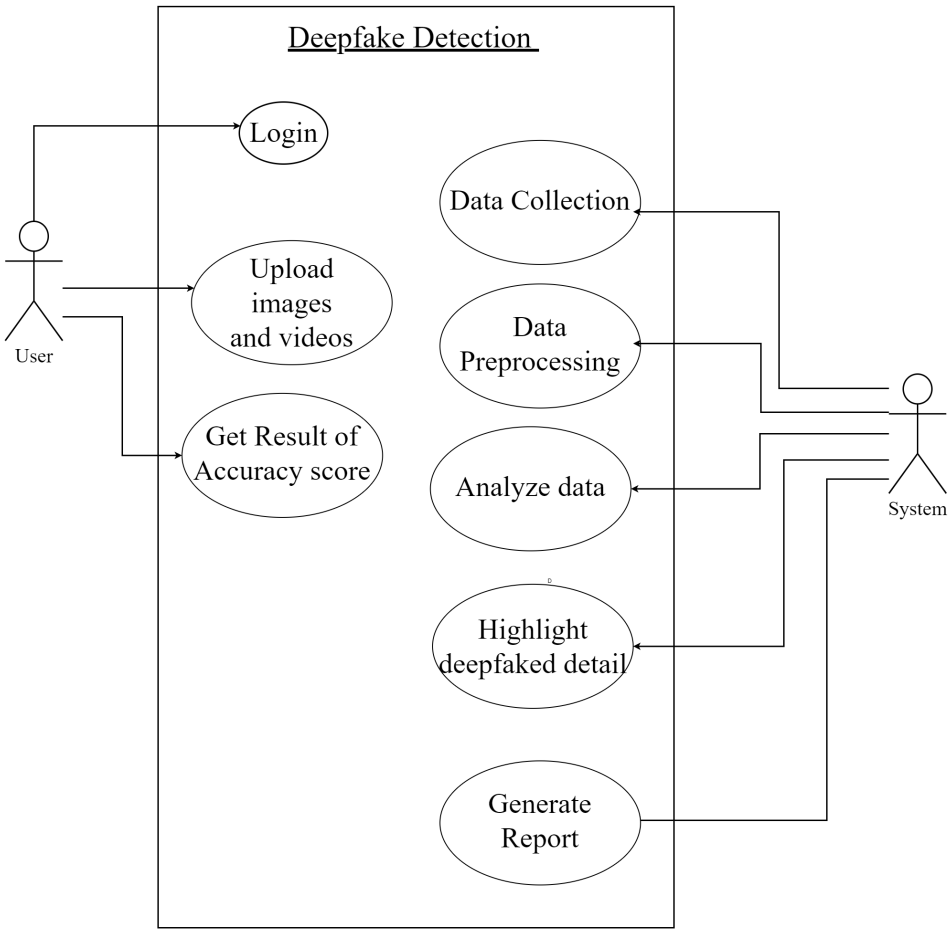

Figure 8: Use Case Diagram

The use case diagram for our system illustrates various interactions and roles of the system's users. The primary actors involved are the "User" and the "System." The User interacts with the system by initiating the deepfake detection process, either by uploading a video or an image. The User can also access the system to view the detection results. On the other hand, the System is responsible for managing the system, including user authentication, system configuration, and monitoring the overall functionality. The use case diagram shows the main use cases, such as "Upload Media," "Detect Deepfake," and "View Results," which represent the key functionalities of the system.

5.3 Sequence Diagram

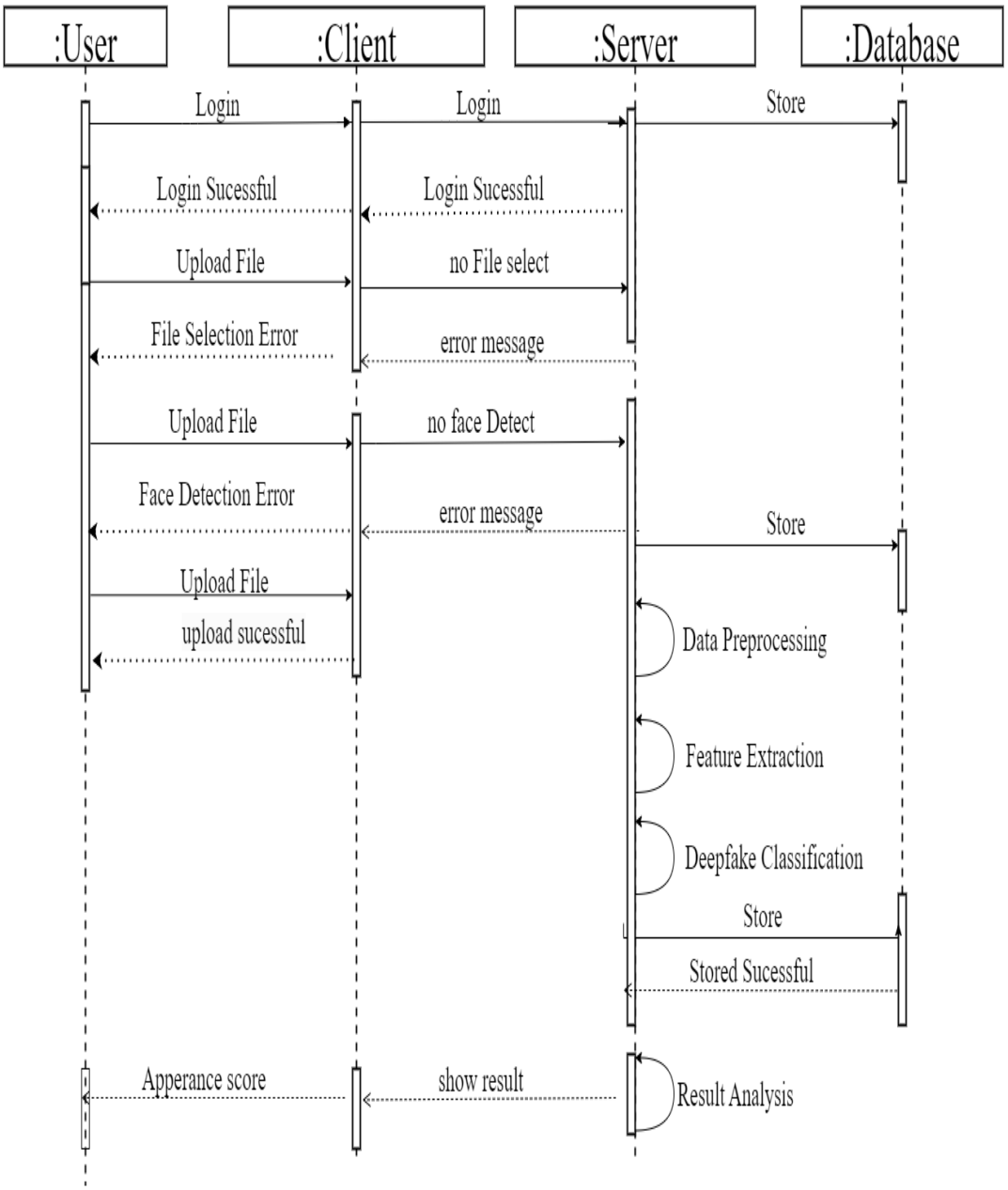

Figure 9: Sequence Diagram

The sequence diagram shows that the user initiates the process by accessing the system and providing their login credentials. The Server checks the login credentials and verifies the user's identity. Once authenticated, the user proceeds to upload a file containing the video or image to be analyzed for deepfakes.Then face detection algorithms is used to detect and extract faces from the uploaded media. This collected data undergoes further processing, including resizing and normalization, to prepare it for deep learning modeling. Finally, the processed data is fed into the deep learning model, which analyzes the features and patterns to classify the media as either real or fake.

5.4 Dataflow Diagram

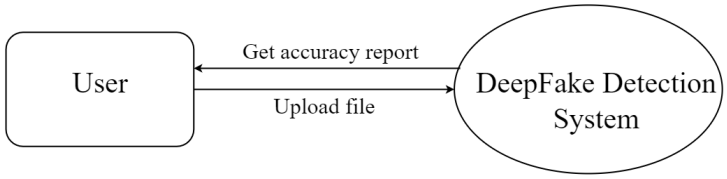

Figure 10: Level 0 DFD

DFD level – 0 indicates the basic flow of data in the system.

- User: User input to the system is uploading video.
- System: In system it shows all the details of the Video and output shows the fake video or not.
  and output flow

Hence, the data flow diagram indicates the visualization of system with its input feed to the system by User.

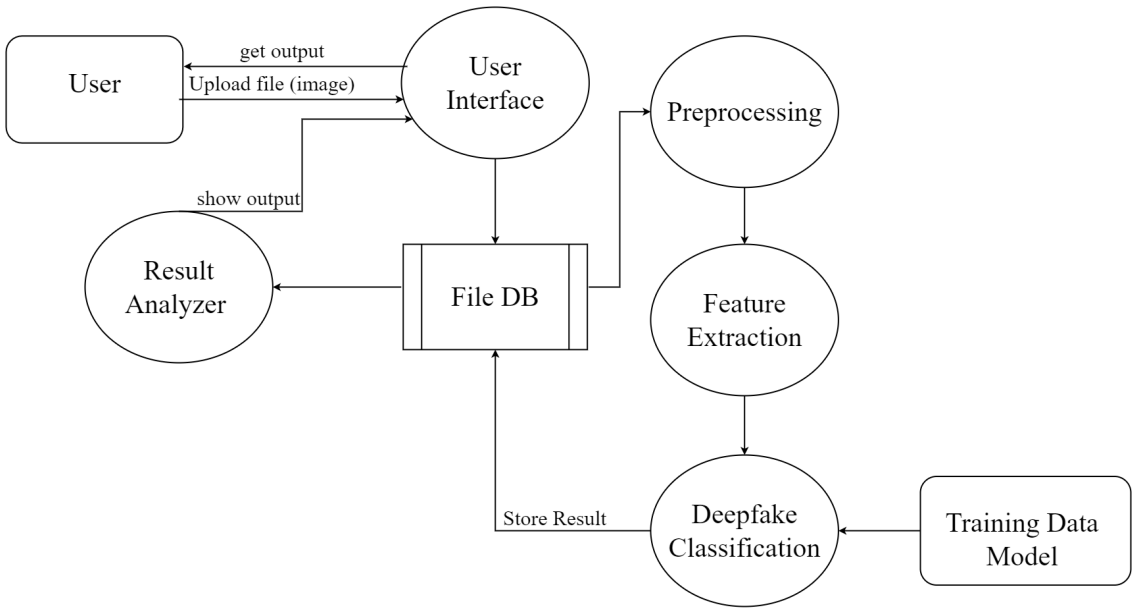

Figure 11: Level 1 DFD

DFD Level – 1 gives more in and out information of the system.Where system gives detailed information of the procedure taking place.

5.5 Activity Diagram

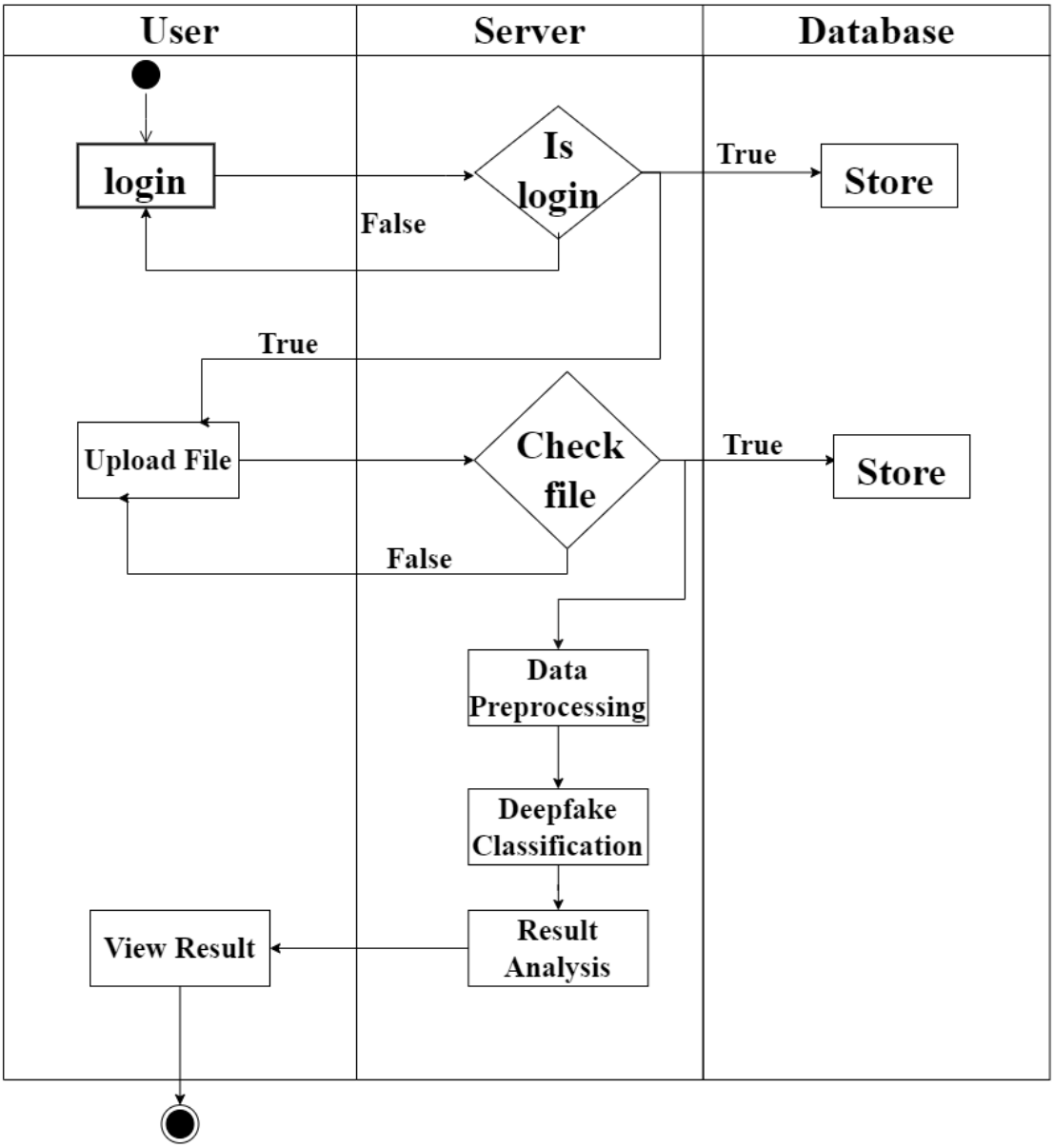

Figure 12: Activity Diagram

The activity diagram shows that the user initiates the process by accessing the system and providing their login credentials. The Server checks the login credentials and verifies the user's identity. Once authenticated, the user proceeds to upload a file containing the video or image to be analyzed for deepfakes.

6 EXPECTED OUTCOMES

- User-friendly interface for easy upload and clear result presentation.
- Accurate identification of manipulated media content.
- Robust performance against different deepfake techniques and adversarial attacks.

References

[1] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies,Matthias Nießner, “FaceForensics++: Learning to Detect Manipulated Facial Images”

[2] Deepfake detection challenge dataset : https://www.kaggle.com/c/deepfake-detection challenge/data

[3] Yuezun Li , Xin Yang , Pu Sun , Honggang Qi and Siwei Lyu “Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics”

[4] 10 deepfake examples that terrified and amused the internet : https://www.creativebloq.com/features/deepfake-examples

[5] Keras: https://keras.io/

[6] PyTorch : https://pytorch.org/

[7] G. Antipov, M. Baccouche, and J.-L. Dugelay. Face aging with conditional generative adversarial networks.

[8] TensorFlow: https://www.tensorflow.org/

[9] Face app: https://www.faceapp.com/

[10] Face Swap : https://faceswaponline.com/