TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

LALITPUR
ENGINEERING COLLEGE
(Affiliated to Tribhuvan University)

LALITPUR ENGINEERING COLLEGE
KHOLKA POKHARI, LALITPUR

A PROPOSAL OF MAJOR PROJECT

DefaceLab: DeepFake Detection using Deep Learning

SUBMITTED BY

ABHISHEK NEUPANE (LEC-076-BCT-02)
RABINDRA ADHIKARI (LEC-076-BCT-025)
SANJISH MAHARJAN (LEC-076-BCT-032)
SUSHIL KAFLE (LEC-076-BCT-045)

SUBMITTED TO

DEPARTMENT OF COMPUTER ENGINERING

2080-02-25

DefaceLab: DeepFake Detection using Deep Learning

Submitted by
ABHISHEK NEUPANE (LEC-076-BCT-02)
RABINDRA ADHIKARI (LEC-076-BCT-025)
SANJISH MAHARJAN (LEC-076-BCT-032)
SUSHIL KAFLE (LEC-076-BCT-045)

Project Supervisor
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

CNN	CSR
RNN	CRR
	DMG
DUDNC	DUDNC
EERI	EERI

1 INTRODUCTION

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 (i.e., realistic-looking fake media that has been generated by deep-learning algorithms) 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].

The generation of deepfake media can have a wide range of intentions and motivations, from revenge porn to political fake news.. Deepfakes have also been published to falsify satellite images with non-existent landscape features for malicious purposes [3]. There are numerous captivating applications of deepfakery in video compositing and transfiguration in portraits, especially in identity protection as it can replace faces in photographs with ones from a collection of stock images. Cyber-attackers, using various strategies other than deepfakery, are always aiming to penetrate identification or authentication systems to gain illegitimate access. Therefore, identifying deepfake media using forensic methods remains an immense challenge since cyber-attackers always leverage newly published detection methods to immediately incorporate them in the next generation of deepfake generation methods. With the massive usage of the Internet and social media, and billions of images available on the Internet, there has been an immense loss of trust from social media users. Deepfakes are a significant threat to our society and to digital evidence in courts. Therefore, it is highly important to obtain state-of-the-art techniques to identify deepfake media under criminal investigation. As demonstrated in Table 1 (inspired by the figure presented in [1]), tampering of evidence, scams and frauds (i.e., fake news), digital kidnapping associated with ransomware blackmailing, revenge porn and political sabotage are among the vast majority of types of deepfake activities with the highest level of intention to mislead [1].

1.1 Background

1.2 Problem Statement

1.3 Objectives

- Our project aims at discovering the distorted truth of the deep fakes.
- Our project will reduce the Abuses' and misleading of the common people on the world wide web.
- Our project will distinguish and classify the video as deepfake or pristine.
- Provide a easy to use system for used to upload the video and distinguish whether the video is real or fake

1.4 Scope

Our deepfake detection project aims to provide a web-based platform for users to upload videos and classify them as either fake or real. The platform will serve as a valuable tool in preventing the proliferation of deepfakes on the internet. Additionally, we envision the potential to scale up the project by developing a browser plugin for automatic deepfake detection. This would enable seamless integration with popular applications such as WhatsApp and Facebook, allowing users to detect deepfakes before sharing them with others. The system will support video uploads of varying sizes, ranging from small video clips to larger files. The exact size limit will depend on the implementation and infrastructure constraints. However, efforts will be made to accommodate a wide range of video sizes to ensure user convenience.

2 LITERATURE REVIEW

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

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.

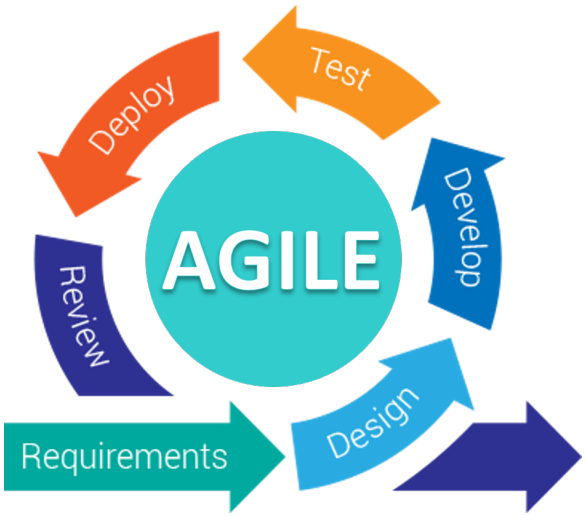

Figure 3: Agile Model

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

5 BLOCK DIAGRAMS

5.1 System Architecture

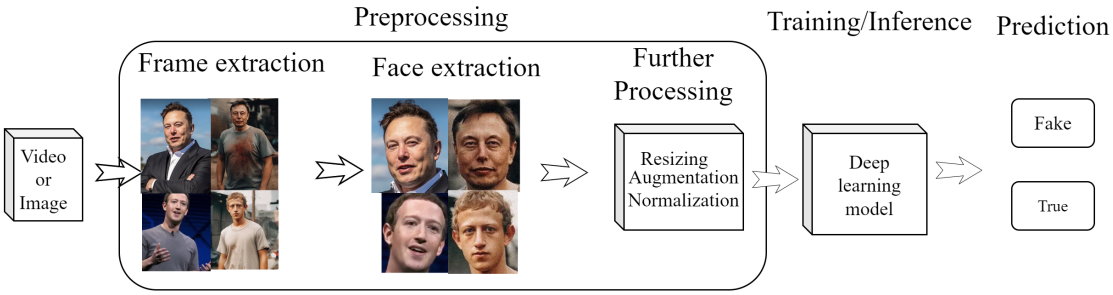

Figure 4: System Architecture

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

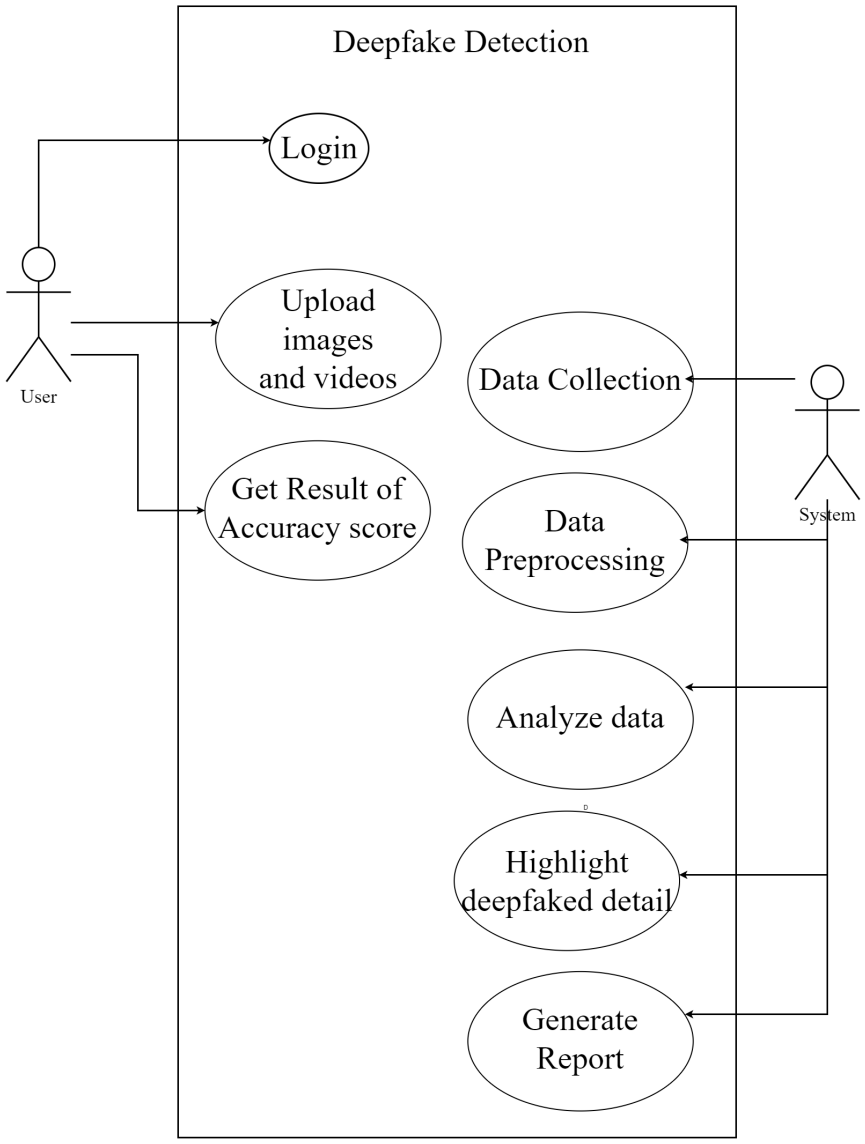

Figure 5: Use Case Diagram

5.3 Sequence Diagram

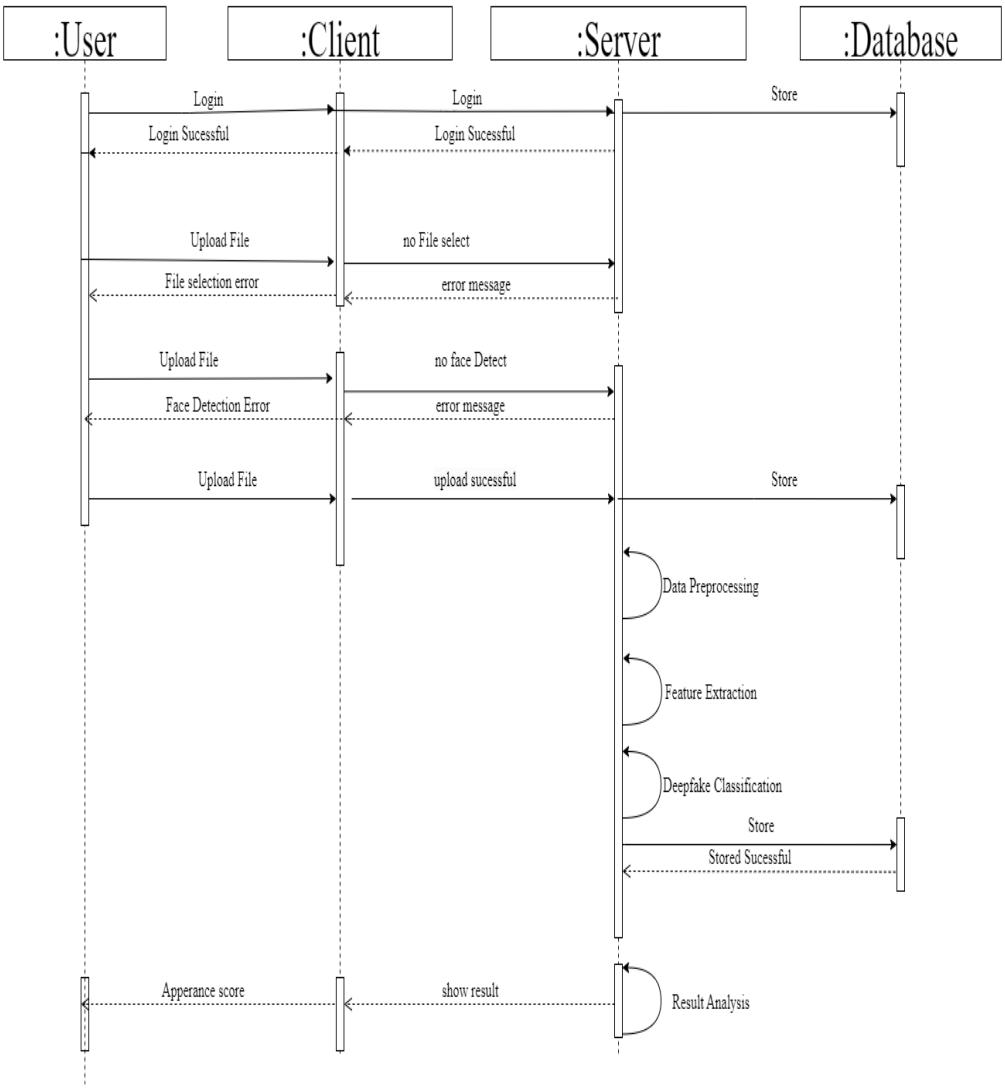

Figure 6: Sequence Diagram

5.4 Dataflow Diagram

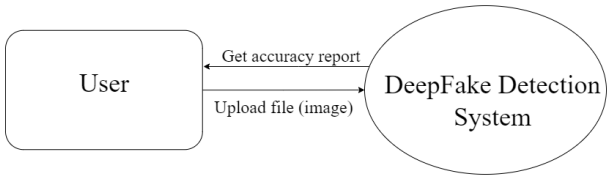

Figure 7: Level 0 DFD

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.

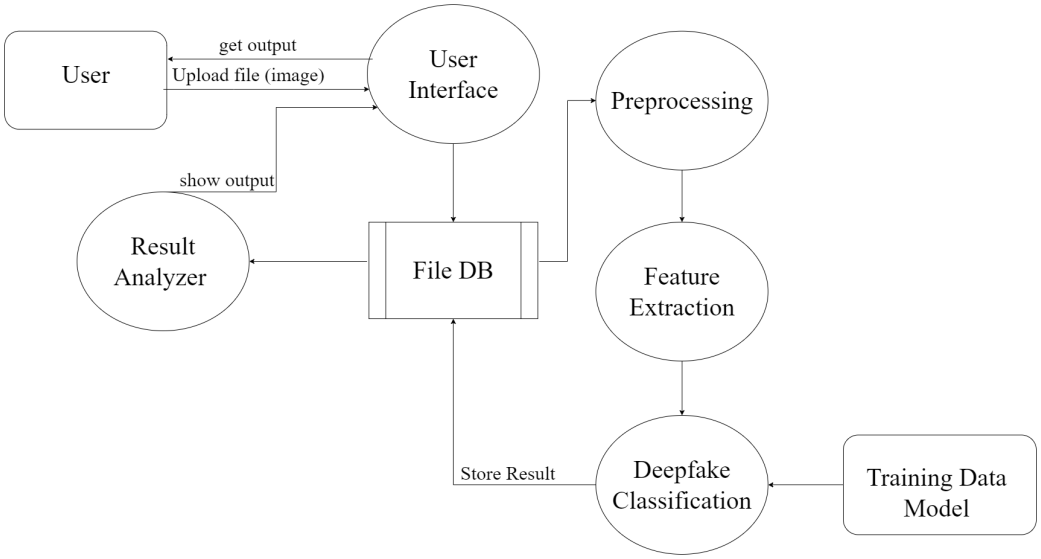

Figure 8: Level 1 DFD

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

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.

7 LIMITATIONS

- Deepfake detection projects face challenges due to rapidly evolving techniques and the need for diverse training data.
- Adversarial attacks can exploit weaknesses in detection algorithms, making deepfakes harder to identify accurately.
- Deepfake detection algorithms often require significant computational resources, limiting their applicability on resource-constrained devices.

8 FUTURE ENHANCEMENTS

There is always a scope for enhancements in any developed system, especially when the project build using latest trending technology and has a good scope in future.

- Web based platform can be upscaled to a browser plugin for ease of access to the user.
- Currently only Face Deep Fakes are being detected by the algorithm, but the algorithm can be enhanced in detecting full body deep fakes.

References

[1] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies,Matthias Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images" in arXiv:1901.08971.

[2] Deepfake detection challenge dataset : https://www.kaggle.com/c/deepfake-detection challenge/data Accessed on 26 March, 2020

[3] Yuezun Li , Xin Yang , Pu Sun , Honggang Qi and Siwei Lyu "Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics" in arXiv:1909.12962

[4] 10 deepfake examples that terrified and amused the internet : https://www.creativebloq.com/features/deepfake-examples Accessed on 26 March, 2020

[5] Keras: https://keras.io/ (Accessed on 26 March, 2020)

[6] PyTorch : https://pytorch.org/ (Accessed on 26 March, 2020)

[7] G. Antipov, M. Baccouche, and J.-L. Dugelay. Face aging with conditional generative adversarial networks. arXiv:1702.01983, Feb. 2017

[8] TensorFlow: https://www.tensorflow.org/ (Accessed on 26 March, 2020)

[9] Face app: https://www.faceapp.com/ (Accessed on 26 March, 2020)

[10] Face Swap : https://faceswaponline.com/ (Accessed on 26 March, 2020)