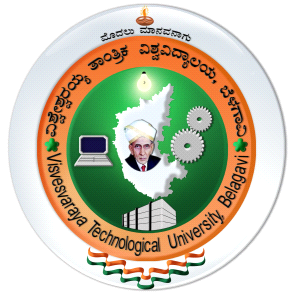
**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**BELAGAVI, KARNATAKA - 590018**



Software Requirements Specification of

**“Deep-Fake Detection”**

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

**Bachelor of engineering**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

By

**Calvin Salvador Dsouza 4SO17CS023**

**G Sunil 4SO17CS039**

**Harsh Pawaskar 4SO17CS041**

**Milton Rodrigues 4SO17CS068**

Under the guidance of

**Smitha V George**

Assistant Professor

Department of Computer Science and Engineering



**ST JOSEPH ENGINEERING COLLEGE**

**MANGALURU 575028**

**2020-21**

1. **Introduction**

**1.1 Purpose**

In recent years, various social issues arise because of fake videos called DeepFake. These deepfake videos are computer-manipulated images in which one person's likeness has been used to replace that of another. The main purpose of this application is to find out the whether video is real or fake and which may go to be viral in social media platform.

**1.2 Product Scope**

Deepfake detection is application-based project to differentiate videos are real or fake using convolutional neural network. Although the current market uses different gan techniques. But we focus on the detection of the lack of eye blinking to expose AI synthesize face videos. We calculate the overall rate of blinking, and compares that with the natural range. This technique helps us to increases the accuracy rate of the output.

**1.3 Overview**

1. The user will upload the video into the application and converted into stand form of 24fps.
2. Target-Detector performs the function of identifying the face and landmark localization which is then forwarded to the Eye-Tracker.
3. Eye-Tracker identifies the eye and tracks the blinks of the eyes.
4. Data computed from the Eye-tracker is compared with the database to find out real or fake videos.
5. **Overall Description**

**2.1 Product Perspective**

Deepfake detector is a system-based application product which provides the functionality of identifying deepfake videos. The main focus is to detect the lack of physiological signals intrinsic to human beings that are not well captured in the synthesized videos. Application determines deepfakes through integrity verification by tracking significant changes in the eye blinking patterns in deepfakes by means of a heuristic method based on the results of biology and brain engineering research, as well as machine learning and various algorithms based on engineering and statistical knowledge.

**2.2 Product Functions**

* **GUI Interaction:** Allows a user to upload a video file to the application for processing.
* **Help:** Application guides the user through the entire process of deepfake detection by providing a visual and text file which gives demo of the applications working.
* **Pre-processing:** The application performs basic validity checks at the pre-processing phase to make sure the user has uploaded the file of size and format which adheres to the application specified format.
* **Face-detection:** Performs face detection and landmark localization. In a scenario where the user uploads a file with no individual present or video with very low illumination the application informs the user regarding the same.
* **Eye-Tracker:** Performs eye detection and tracks blinking of the eye.
* **Measurements and Analysis:** Data computed from the Eye-tracker is compared with the database to evaluate whether the video is artificially synthesized using ai or not.

**2.3 User Characteristics**

**General Users:** Any individual who wants to verify the integrity of a video can utilize the application to identify if the video is a synthetic file produced by sophisticated artificial intelligence.

**Cyber-crime Employees:** Members of the cyber-security can employ this software to protect privacy of any individual and also identify the deepfake videos used to defame popular celebrities and politicians.

**2.4 Specific Constraints**

* The file uploaded should be in .mp4 format.
* File size should be less than 100mb.
* The video file should be clear with appropriate illumination

**2.5 General Constraints**

**• Hardware**: Laptop or PC with Processor x86\_64 and minimum 8 Gb of RAM.

• **Python version**: Python 3.7 and above.

• **GPU**: Minimum 4 GB graphic card (NVIDIA preferred).

• **GUI**: Is Available only in English.

• **Operating System**: Windows XP and above, Mac OS and Linux.• **Storage**: System with minimum 10GB of free space to store the datasets.

**2.6 Assumptions and Dependencies**

* The Efficiency of application will have a drastic dropdown if the user creates and uploads a video with either no blinks or rapid blinks on purpose in which case the application might fail.
* The software does not take into consideration the activities such as yawning, sneezing, or winking which might interfere with its accuracy of results.

1. **Specific Requirements**

**3.1 External Interface Requirements**

**User Interface**

* Video Upload Page: Video upload feature which enables users to upload video for processing.
* Result page: A page which displays the result stating the probability of being a deepfake video.

**Hardware Interface**

Device with screen and internet to access the website.

**Software Interface**

* Operating system: Any operating system with python installed is supported.
* Client on the Internet - Web browser
* Database: MongoDB.
* Programming language: Python3.
* Video player - Windows Media player Plug-in.

**Communication Interface**

Device with internet connectivity and browser.

**3.2 Functional Requirements**

* Uploading videos:

The application shall provide an interface to upload Deep Fake or Real videos into server.

* Detecting the Deep Fake Videos:

The application would process the videos uploaded by users to detect whether it is a Deep Fake or Real videos.

**3.3 Non-Functional Requirements**

* Performance Requirements:

The system shall process a minimum of 26 video frames per second. Performance of the system may vary based on the quality of video being uploaded.

* Availability:

The system should be available at all times since it is hosted on a web server.

**3.4 Design Constraints**

* The graphical user interface should have a consistent look and feel. The image processing should be effective enough to do its detection in a minimum amount of time.
* Since the users are required to upload videos into server, a proper internet connection is required.