SOFTWARE REQUIREMENT SPECIFICATION (SRS)

Deepfake Detection Using Machine Learning

1. Introduction

This Software Requirement Specification document provides an overview of the functionality of Deepfake Detection System inorder to detect fake images and videos that cannot be detected using our eyes. This document will cover the scope, organization, description, constraints and requirements of Deepfake Detection.

1.1 Purpose

The purpose of this document is to describe the functionality and specifications of the design of Deepfake Detection System. The expected audiences of this document are the developers and users of this software.

1.2 Scope

This deepfake detection system is designed to run on computers and laptops. The user will be able to upload images and videos that they want to check. The software checks the images and videos and extracts its features. Those features are compared with the features of fake data (that the software learned during training phase) and returns whether it is fake or not.

1.3 Definitions, acronyms and abbreviation

- CNN(Convolutional Neural Network) : In deep learning, a convolutional neural network is a class of deep neural neworks, most commonly applied to analyzing visual imagery.

- RNN(Recurrent Neural Network) : They are type of neural network where the output from the previous step are fed as input to the current step.

- DF(Deep Fake) : They are synthetic media in which a person in an existing image or video is replaced with someone else's likeness.

- AI(Artificial Intelligence) : It refers to simulation of human intelligence in machines that are programmed to think like humans and mimic their actions

- ML(Machine Learning) : It is the study of computer algorithms that improve automatically through experience.

- DL(Deep Learning) : It is a part of broader family of machine learning methods based on artificial neural networks with representation learning.

- GAN(Generative Adversarial Network): They are algorithimic architectures that use two neural networks, pitting against the other in order to generate new, synthetic instances of data that can pass for real data.

- LSTM(Long Short-Term Memory) : It is an artificial recurrent neural network architecture used in the field of deep learning.

- ROI(Region of Interest) : They are samples within a dataset identified for a particular purpose.

1.4 Organization

The remaining portions of this document are decomposed into three major sections followed by references. The first section will provide overall description of this project and the next part will give more detailed requirements including software and hardware specifications, functional and non-functional parameters etc.

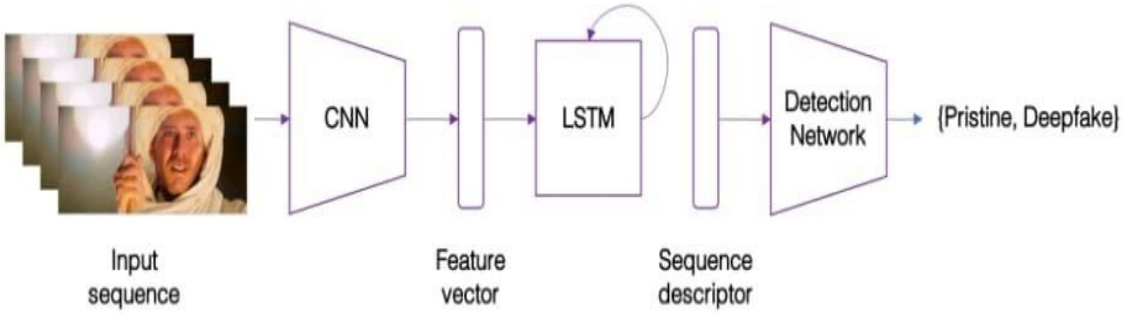
2 Overall description

This section provides a high level description of the entire software. It describes product perspective, functionality and characteristics of expected user, constraints, assumptions and dependencies, future requirements .

2.1 Product Perceptive

Deepfakes are increasingly detrimental to privacy, society security and democracy. This software is specifically designed to detect DF. Two sets of training images are required. The first set only has samples of the original face that will be replaced, which can be extracted from the target video that will be manipulated. The second set of images contains the desired face that will be swapped in the target video. To ease the training process of the autoencoders, the easiest face swap would have both the original face and target face under similar viewing and illumination conditions. The proposed system is composed by a convolutional LSTM structure for

processing frame sequences. There are two essential components in a convolutional LSTM. CNN for frame extraction and LSTM for temporal sequence analysis.



The system learns and infers in an end-to-end manner and, given a video sequence, outputs a probability of it being a deepfake or a pristine video. It has a convolutional LSTM subnetwork, for processing the input temporal sequence.

2.2 Product Functions

- Detects whether a video or image is fake or not with more accuracy and cost effectively.
- Helps in preventing the usage of deepfakes in creating political distress, blackmailing, fake terrorism etc.
- Protection of identity and privacy of person.

2.3User characteristics

- The user should be familiar with basic functionality of laptops and computers.
- User should know how to upload videos and pictures in laptop/computer inorder to input data in the software.
- User should know to run a computer program. Programming skills are not necessary for the user.

2.4 Constraints

- As autoencoder is using frame by frame, it is completely unaware of any previous generated face that it may have created. This lack of temporal awareness is the source of multiple anomalies.

- Too smooth skin and lack of skin details are consequence of one problem in deepfake algorithms.
- Requires large amounts of real and fake dataset for training for better experience and accuracy. But if new deepfake generation methods are introduced, it is challenging to collect large amounts of fake data in advance.

2.5 Assumptions and Dependencies

Even though ML is a complex discipline , implementing machine learning models has become less difficult than it used to be as we are implementing in Google's Tensorflow as it eases the process of acquiring the data, training models,serving predictions and refining future results. Tensorflow is a python-friendly open source library for numerical computation. So we can assume our software can be implemented well in tensorflow using python as python offers concise and readable code. While complex algorithms and versatile workflows stand behind machine learning , Python's simplicity allows developers to write reliable code. Tensorflow works pretty well in both linux and windows 10 . But as linux has all the languages and dependencies already installed in its core , we can assume it is better to use linux for development of software.

2.6 Future requirements

One of the things the user would like to implemented is a more robust application in computer as well as that in phone. So that it would be more user friendly as well as more accurate even for unseen manipulated videos.

3 Specific Requirements

This section provides further details of our software. It includes hardware and software requirements needed for the development of this software.

3.1 Hardware Specifications

As we are detecting deepfakes using machine learning, its building process can be intimidating and time consuming. So the minimum hardware requirement that we would need are

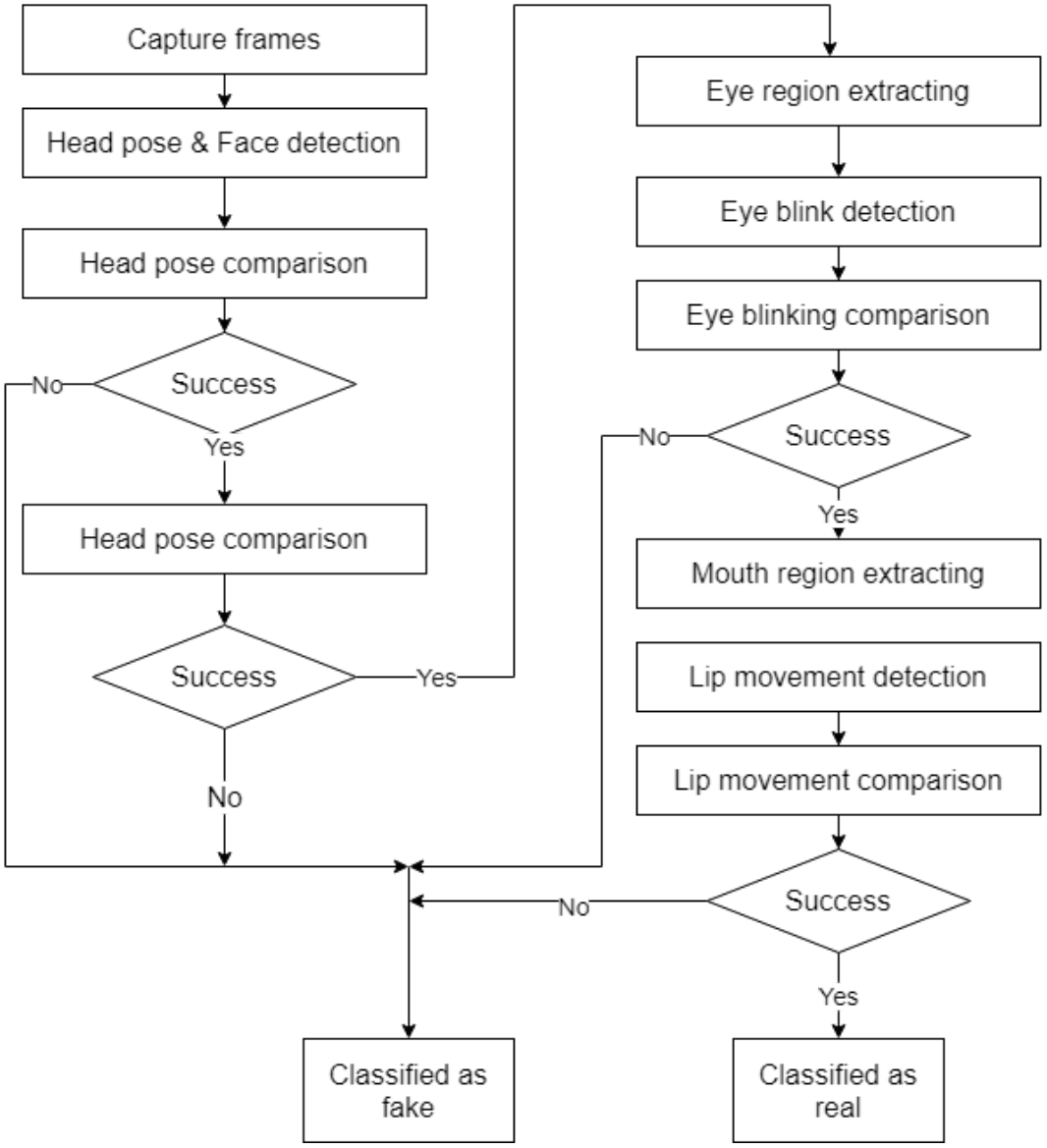
- CPU : Intel Core i5 or above
- RAM : 4GB
- Hard Disk : 500 GB

- Decent wireless or Ethernet connection to download/upload datasets for practice.

3.2 Software specifications

- Operating System : Windows
- Programming Platform : VS Code / Jupyter Notebook / Google Colab
- Programming Technologies : Python, Tensorflow

4 System Working Model



This is the Level 2 DFD diagram of our software. Dataset consisting of videos are fed to the detector and firstly frames are captured from the videos. Headpose, eye region and lip movement are analyzed and compared with trained dataset and outputs whether those videos are fake or not.