DEEPFAKE DETECTION USING DEEP LEARNING

A SUMMER INTERNSHIP PROJECT REPORT

Submitted by

SOWMITHA. M (2022115007)

JAYASHREE. J (2022115010)

SIVASANGARI. S (2016108040)

COLLEGE OF ENGINEERING GUINDY, ANNA UNIVERSITY

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2024

BONAFIDE CERTIFICATE

Certified that this project report “DEEPFAKE DETECTION USING DEEP LEARNING ” is the bonafide work of “SOWMITHA.M (2022115007), JAYASHREE.J (2022115010), SIVASANGARI.S (2022115)” who carried out the project work under my supervision.

Dr. SWAMYNATHAN, M.E., Ph.D.,
Professor and Head
Department of Information Science and Technology
College of Engineering Guindy
Anna University, Chennai-25

Dr. R. BASKARAN, M.E.,Ph.D.,
Associate Professor and Guide
Department of Information Science and Technology
College of Engineering Guindy
Anna University, Chennai-25

ACKNOWLEDGMENT

We would firstly like to express our deep gratitude to our Professor, guide and mentor, Dr. R. BASKARAN, Associate Professor, and Department of Industrial Engineering, who guided us throughout the project and gave useful tips until the end. We express our sincere thanks to the Dr. P. HARIHARAN, Professor and Head of the Department, Department of Manufacturing Engineering for his guidance and support right from the initial stages of the project. We also express our gratitude to Dr. K. SELVAMANI, Assistant Professor, Department of Computer Science and Engineering for his guidance and support till the end of the project.

We would also like to thank Dr. R. RAJU, Professor and Head, Department of Industrial Engineering for his encouragement in the proceedings of this project.

We finally would like to acknowledge and thank, the support received from the other teaching and non-teaching faculty members of the department.

Sowmitha. M	Jayashree.J	Sivasangari.S
(2022115007)	(2022115010)	(2022115114)

ABSTRACT

The growing computation power has made the deep learning algorithms so powerful that creating a indistinguishable human synthesized video popularly called as deep fakes have became very simple. Scenarios where these realistic face swapped deep fakes are used to create political distress, fake terrorism events, revenge porn, blackmail peoples are easily envisioned. In this work, we describe a new deep learning-based method that can effectively distinguish AI-generated fake videos from real videos.Our method is capable of automatically detecting the replacement and reenactment deep fakes. We are trying to use Artificial Intelligence(AI) to fight Artificial Intelligence(AI). Our system uses a Res-Next Convolution neural network to extract the frame-level features and these features and further used to train the Long Short Term Memory(LSTM) based Recurrent Neural Network(RNN) to classify whether the video is subject to any kind of manipulation or not, i.e whether the video is deep fake or real video. To emulate the real time scenarios and make the model perform better on real time data, we evaluate our method on large amount of balanced and mixed data-set prepared by mixing the various available data-set like Face-Forensic++[1], Deepfake detection challenge[2], and Kaggle[3]. We also show how our system can achieve competitive result using very simple and robust approach.

Keywords:

Res-Next Convolution neural network.

Recurrent Neural Network (RNN).

Long Short Term Memory(LSTM)

Table of Contents

LIST OF TABLES

Contents

CHAPTER 1

INTRODUCTION

1.1 Project Idea

In the world of ever growing Social media platforms, Deepfakes are con sidered as the major threat of the AI. There are many Scenarios where these realistic face swapped deepfakes are used to create political distress, fake ter rorism events, revenge porn, blackmail peoples are easily envisioned.Some of the examples are Brad Pitt, Angelina Jolie nude videos.

It becomes very important to spot the difference between the deepfake and pris tine video. We are using AI to fight AI.Deepfakes are created using tools like FaceApp[11] and Face Swap[12], which using pre-trained neural networks like GANorAutoencoders for these deepfakes creation. Our method uses a LSTM based artificial neural network to process the sequential temporal analysis of the video frames and pre-trained Res-Next CNN to extract the frame level fea tures. ResNext Convolution neural network extracts the frame-level features and these features are further used to train the Long Short Term Memory based artificial Recurrent Neural Network to classify the video as Deepfake or real. To emulate the real time scenarios and make the model perform better on real time data, we trained our method with large amount of balanced and combi nation of various available dataset like FaceForensic++[1], Deepfake detection challenge[2], and Kaggle[3].

Further to make the ready to use for the customers, we have developed a front end application where the user the user will upload the video. The video will be processed by the model and the output will be rendered back to the user with the classification of the video as deepfake or real and confidence of the model

1.2 Motivation of the Project

The increasing sophistication of mobile camera technology and the ever growing reach of social media and media sharing portals have made the creation and propagation of digital videos more convenient than ever before. Deep learning has given rise to technologies that would have been thought impossible only a handful of years ago. Modern generative models are one example of these, capable of synthesizing hyper realistic images, speech, music, and even video. These models have found use in a wide variety of applications, including making the world more accessible through text-to-speech, and helping generate training data for medical imaging.Like anytrans-formative technology, this has created new challenges. So-called "deep fakes" produced by deep generative models that can manipulate video and audio clips. While many are likely intended to be humorous, others could be harmful to individuals and society. Until recently, the number of fake videos and their degrees of realism has been increasing due to availability of the editing tools, the high demand on domain expertise.

Spreading of the Deep fakes over the social media platforms have become very common leading to spamming and peculating wrong information over the plat form. Just imagine a deep fake of our prime minister declaring war against neighboring countries, or a Deep fake of reputed celebrity abusing the fans. These types of the deep fakes will be terrible, and lead to threatening, mislead ing of common people.

To overcome such a situation, Deep fake detection is very important. So, we describe a new deep learning-based method that can effectively distinguish AI generated fake videos (Deep Fake Videos) from real videos. It’s incredibly important to develop technology that can spot fakes, so that the deep fakes can be identified and prevented from spreading over the internet.

CHAPTER 2

Literature Survey

Face Warping Artifacts [15] used the approach to detect artifacts by com paring the generated face areas and their surrounding regions with a dedicated Convolutional Neural Network model. In this work there were two-fold of Face Artifacts.

Their method is based on the observations that current deepfake algorithm can only generate images of limited resolutions, which are then needed to be fur ther transformed to match the faces to be replaced in the source video. Their method has not considered the temporal analysis of the frames.

Detection by Eye Blinking [16] describes a new method for detecting the deep fakes by the eye blinking as a crucial parameter leading to classification of the videos as deepfake or pristine. The

Long-term Recurrent Convolution Network (LRCN) was used for temporal analysis of the cropped frames of eye blinking. Astodaythedeepfake generation algorithms have become so powerful that lack of eye blinking cannot be the only clue for detection of the deepfakes. There must be certain other parameters must be considered for the detection of deep fakes like teeth enchantment, wrinkles on faces, wrong placement of eyebrows etc.

Capsule networks to detect forged images and videos [17] uses a method that uses a capsule network to detect forged, manipulated images and videos in different scenarios, like replay attack detection and computer-generated video detection.

In their method, they have used random noise in the training phase which is not a good option. Still the model performed beneficial in their dataset but may fail on real time data due to noise in training. Our method is proposed to be trained on noiseless and real time datasets.

Recurrent Neural Network [18] (RNN) for deepfake detection used the ap proach of using RNN for sequential processing of the frames along with ImageNet pre-trained model. Their process used the HOHO [19] dataset consisting of just 600 videos.

Their 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.

Synthetic Portrait Videos using Biological Signals [20] approach extract bio logical signals from facial regions on pristine and deepfake portrait video pairs. Applied transformations to compute the spatial coherence and temporal consistency, capture the signal characteristics in feature vector and photoplethysmography (PPG) maps, and further train a probabilistic Support Vector Machine (SVM) and a Convolutional Neural Network (CNN). Then, the average of authenticity probabilities is used to classify whether the video is a deepfake or a pristine.

Fake Catcher detects fake content with high accuracy, independent of the generator, content, resolution, and quality of the video. Due to lack of discriminator leading to the loss in their findings to preserve biological signals, formulating a differentiable loss function that follows the proposed signal processing steps is not straight forward process.

Chapter 3
Problem Definition and scope

3.1 Problem Statement

Convincing manipulations of digital images and videos have been demonstrated for several decades through the use of visual effects, recent advances in deep learn ing have led to a dramatic increase in the realism of fake content and the accessibility in which it can be created. These so-called AI-synthesized media (popularly referred to as deep fakes).Creating the Deep Fakes using the Artificially intelligent tools are simple task. But, when it comes to detection of these Deep Fakes, it is major chal lenge. Already in the history there are many examples where the deepfakes are used as powerful way to create political tension[14], fake terrorism events, revenge porn, blackmail peoples etc.So it becomes very important to detect these deepfake and avoid the percolation of deepfake through social media platforms. We have taken a step forward in detecting the deep fakes using LSTM based artificial Neural network.

3.1.1 Goals and objectives

Goal and 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.

3.1.2 Statement of scope

There are many tools available for creating the deep fakes, but for deep fake detection there is hardly any tool available. Our approach for detecting the deep fakes will be great contribution in avoiding the percolation of the deep fakes over the world wide web. We will be providing a web-based platform for the user to upload the video and classify it as fake or real. This project can be scaled up from developing a web-based platform to a browser plugin for au tomatic deep fake detection's. Even big application like WhatsApp, Facebook can integrate this project with their application for easy pre-detection of deep fakes before sending to another user. A description of the software with Size of input, bounds on input, input validation, input dependency, i/o state diagram, Major inputs, and outputs are described without regard to implementation detail.

CHAPTER 4
PROBLEM IDENTIFICATION

4.1 Major Constraints User:

- User of the application will be able detect the whether the uploaded video is fake or real, Along with the model confidence of the prediction.
- Prediction: The User will be able to see the playing video with the output on the face along with the confidence of the model.
- Easy and User-friendly User-Interface: Users seem to prefer a more sim plified process of Deep Fake video detection. Hence, a straight forward and user-friendly interface is implemented.The UI contains a browse tab to select the video for processing. It reduces the complications and at the same time enrich the user experience.
- Cross-platform compatibility: with an ever-increasing target market, accessibility should be your main priority. By enabling a cross-platform compatibility feature, you can increase your reach to across different platforms.

Being a server side application it will run on any device that has a web browser installed in it.

4.2 Methodologies of Problem solving

4.2.1 Analysis

- Solution Requirement

  We analysed the problem statement and found the feasibility of the solution of the problem. We read different research paper as mentioned in 3.3. After checking the feasibility of the problem statement. The next step is the data set gathering and analysis. We analysed the data set in different approach of training like negatively or positively trained i.e training the model with only fake or real video's but found that it may lead to addition of extra bias in the model leading to inaccurate predictions. So after doing lot of research we found that the balanced training of the algorithm is the best way to avoid the bias and variance in the algorithm and get a good accuracy.
- Solution Constraints

  We analysed the solution in terms of cost,speed of processing,requirements,level of expertise, availability of equipment's.
- Parameter Identified

  1.Blinking of eyes 2. Teeth enchantment 3. Bigger distance for eyes 4. Moustaches 5. Double edges, eyes,

ears, nose 6. Iris segmentation 7. Wrinkles on face 8. Inconsistent head pose 9. Face angle 10. Skin tone 11. Facial Expressions 12. Lighting 13. Different Pose 14. Double chins 15. Hairstyle 16. Higher cheek bones

4.2.2 Design

After research and analysis we developed the system architecture of the solution as mentioned in the Chapter 6. We decided the baseline architecture of the Model which includes the different layers and their numbers.

4.2.3 Development

After analysis we decided to use the PyTorch framework along with python3 language for programming. PyTorch is chosen as it has good support to CUDA i.e Graphic Processing Unit (GPU) and it is customize-able. Google Cloud Platform for training the final model on large number of data-set.

4.2.4 Evaluation

We evaluated our model with a large number of real time dataset which include YouTubevideos dataset. Confusion Matrix approach is used to evaluate the accuracy of the trained model.

4.3 Outcome

The outcome of the solution is trained deepfake detection models that will help the users to check if the new video is deepfake or real.

4.4 Applications

Web based application will be used by the user to upload the video and submit the video for processing. The model will pre-process the video and predict whether the uploaded video is a deepfake or real video.

4.5 Hardware Resources Required

In this project, a computer with sufficient processing power is needed. This project requires too much processing power, due to the image and video batch processing. Client-side Requirements: Browser: Any Compatible browser device

Sr. No.	Parameter	Minimum Requirement
1	Intel Xeon E5 2637	3.5 GHz
2	RAM	16 GB
3	Hard Disk	100 GB
4	Graphic card	NVIDIA GeForce GTX Titan (12 GB RAM)

Table 4.1: Hardware Requirements

4.6 Software Resources Required

Platform :

1. Operating System: Windows 7+
2. Programming Language : Python 3.0
3. Framework: PyTorch 1.4 , Django 3.0
4. Cloud platform: Google Cloud Platform
5. Libraries : OpenCV, Face-recognition

CHAPTER 5
METHODOLOGY

Algorithm Details :

Preprocessing Details:

- Using glob we imported all the videos in the directory in a python list.
- cv2.VideoCapture is used to read the videos and get the mean number of frames in each video.
- To maintain uniformity, based on mean a value 150 is selected as idea value for creating the new dataset.
- The video is split into frames and the frames are cropped on face location.
- The face cropped frames are again written to new video using Video Writer.
- The new video is written at 30 frames per second and with the resolution of 112 x 112 pixels in the mp4 format.
- Instead of selecting the random videos, to make the proper use of LSTM for temporal sequence analysis the first 150 frames are written to the new video.

Model Details:

The model consists of following layers:

- ResNext CNN : The pre-trained model of Residual Convolution Neural Net work is used. The model name is resnext50_32x4d()[22]. This model consists of 50

layers and 32 x 4 dimensions. Figure shows the detailed implementation of model.

- Sequential Layer : Sequential is a container of Modules that can be stacked together and run at the same time. Sequential layer is used to store feature vector returned by the ResNext model in a ordered way. So that it can be passed to the LSTM sequentially.
- LSTM Layer : LSTM is used for sequence processing and spot the temporal change between the frames.2048-dimensional feature vectors is fitted as the input to the LSTM. We are using 1 LSTM layer with 2048 latent dimensions and 2048 hidden layers along with 0.4 chance of dropout, which is capable to do achieve our objective. LSTM is used to process the frames in a sequential manner so that the temporal analysis of the video can be made, by comparing the frame at ‘t’ second with the frame of ‘t-n’ seconds. Where n can be any number of frames before t.
- ReLU:A Rectified Linear Unit is activation function that has output 0 if the input is less than 0, and raw output otherwise. That is, if the input is greater than 0, the output is equal to the input. The operation of ReLU is closer to the way our biological neurons work. ReLU is non-linear and has the advantage of not having any backpropagation errors unlike the sigmoid function, also for larger Neural Networks, the speed of building models based off on ReLU is very fast.
- Dropout Layer :Dropout layer with the value of 0.4 is used to avoid over f itting in the model and it can help a model generalize by randomly setting the output for a given neuron to 0. In setting the output to 0, the cost

function becomes more sensitive to neighbouring neurons changing the way the weights will be updated during the process of backpropagation.

- Adaptive Average Pooling Layer : It is used To reduce variance, reduce com putation complexity and extract low level features from neighbourhood.2 di mensional Adaptive Average Pooling Layer is used in the model

Model Training Details:

+ Train Test Split:The dataset is split into train and test dataset with a ratio of 70%train videos (4,200) and 30% (1,800) test videos. The train and test split is a balanced split i.e 50% of the real and 50% of fake videos in each split.
+ Data Loader: It is used to load the videos and their labels with a batch size of 4.
+ Training: The training is done for 20 epochs with a learning rate of 1e-5 (0.00001),weight decay of 1e-3 (0.001) using the Adam optimizer
+ Adam optimizer[21]: To enable the adaptive learning rate Adam optimizer with the model parameters is used
+ Cross Entropy: To calculate the loss function Cross Entropy approach is used because we are training a classification problem.
+ Softmax Layer: A Softmax function is a type of squashing function. Squash ing functions limit the output of the function into the range 0 to 1. This allows the output to be interpreted directly as a probability. Similarly, softmax func tions are multi-class sigmoids, meaning they are used in determining probabil ity of multiple classes at once. Since the outputs of a softmax

function can be interpreted as a probability (i.e.they must sum to 1), a softmax layer is typically the final layer used in neural network functions. It is important to note that a softmax layer must have the same number of nodes as the output later. In our case softmax layer has two output nodes i.e REAL or FAKE, also Soft max layer provide us the confidence(probability) of prediction.

+ Confusion Matrix: A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made. Confusion matrix is used to evaluate our model and calculate the accuracy
+ Export Model: After the model is trained, we have exported the model. So that it can be used for prediction on real time data.

Model Prediction Detail:

- The model is loaded in the application.
- The new video for prediction is preprocessed and passed to the loaded model for prediction
- The trained model performs the prediction and return if the video is a real or fake along with the confidence of the prediction.

CHAPTER 6

CONCLUSION:

Conclusion We presented a neural network-based approach to classify the video as deep fake or real, along with the

confidence of proposed model. Our method is capable of predicting the output by processing 1 second of video (10 frames per second) with a good accuracy. We implemented the model by using pre-trained ResNext CNN model to extract the frame level features and LSTM for temporal sequence process ing to spot the changes between the t and t-1 frame. Our model can process the video in the frame sequence of 10,20,40,60,80,100

CHAPTER 7

BIBLIOGRAPHY:

[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] Deepfake Video of Mark Zuckerberg Goes Viral on Eve of House A.I. Hearing : https://fortune.com/2019/06/12/deepfake-mark-zuckerberg/ Accessed on 26 March, 2020

[5] 10 deepfake examples that terrified and amused the internet : https://www.creativebloq.com/features/deepfake-examples Accessed on 26 March, 2020

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

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

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

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

[10] J. Thies et al. Face2Face: Real-time face capture and reenactment of rgb videos. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2387–2395, June 2016. Las Vegas, NV.