iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.68445

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Deepfake Detection Images and Videos Using LSTM and ResNext CNN

Mr. R. Vamsidhar Raju[1], Mr. S. Janakiram[2], Mr. P. Reddy Prasad[3], Mr. B. Lohith[4], Mr. N. Vijaya Kumar[5], Dr. R. Karunia Krishnapriya[6], Mr. V. Shaik Mohammad Shahil[7], Mr. Pandetri Praveen[8]

[1, 2, 3, 4]*UGScholar, Sreenivasa Institute of Technology and Management Studies, Department of CSE, Chittoor, India*

[6]*Associate Professor, Sreenivasa Institute of Technology and Management Studies, Department of CSE, Chittoor, India*

[5, 7, 8]*Assistant Professor, Sreenivasa Institute of Technology and Management Studies, Department of CSE, Chittoor, India*

Abstract: ***The growing power of deep learning algorithms has made creating realistic, AI-generated videos and Images, known as deepfakes, relatively easy. These can be used maliciously to create political unrest, fake terrorism events. To combat this, researchers have developed a deep learning-based method to distinguish AI-generated fake videos from real ones. This method uses a combination of Res-Next Convolution neural networks and Long Short-Term Memory (LSTM) based Recurrent Neural Networks (RNN).***

The Res-Next Convolution neural network extracts frame-level features, which are then used to train the LSTM-based RNN. This RNN classifies whether a video is real or fake, detecting manipulations such as replacement and reenactment deepfakes. To ensure the model performs well in real-time scenarios, it's evaluated on a large, balanced dataset combining various existing datasets like the Deepfake Detection Challenge and Celeb-DF. This approach achieves competitive results using a simple yet robust method.

Keywords: Res-Next Convolution neural network, Convolutional Neural Networks (CNNs), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Computer vision.

I. INTRODUCTION

Deepfakes, or artificially produced media that can trick people into thinking they are real, have become more common as a result of the quick development of deep learning technologies. There are serious risks to national security from deepfakes. Social trust and personal privacy. Therefore, reducing these dangers requires the development of efficient deepfake detection techniques.

A. Problem Statement

Current deepfake detection approaches sometimes depend on outdated computer vision technologies or basic machine learning models, which are readily circumvented by complex deepfake algorithms.

Using the advantages of both recurrent neural networks (RNNs) and convolutional neural networks (CNNs), this study suggests a unique deepfake detection.

B. Proposed Statement

This project introduces a deepfake detection system that blends Long Short-Term Memory (LSTM) networks with the ResNeXt CNN architecture. CNN's ResNeXt is employed to extract spatial characteristics from audio spectrograms or video frames, and the LSTM network examines the relationships and temporal connections between successive audio segments or frames. By utilizing the complementing qualities of CNNs and RNNs, the suggested solution seeks to increase the deepfake detection's accuracy and resilience.

C. Objectives

1) Create a deepfake detection system by fusing LSTM and ResNeXt CNN networks.
2) Assess the suggested system's performance using a reference dataset.
3) Examine the outcomes against the most advanced deepfake detection techniques.
4) Examine how resilient the suggested system is to different kinds of deepfakes and attacks.

D. Expected Outcomes

1) An innovative deepfake detection system that makes use of both CNNs' and RNNs' advantages.
2) Enhanced deepfake detection accuracy and resilience in comparison to current techniques.
3) Information about how well the suggested system defends against different kinds of deepfakes and attacks.

II. LITERATURE REVIEW

Face Warping Artefacts [14] employed a specific Convolutional Neural Network model to compare the generated face areas and their surrounding regions in order to identify artefacts. There were two types of face artefacts in this piece.
Their approach is predicated on the finding that the deepfake algorithm now in use can only produce images with a restricted resolution, which must thereafter undergo additional processing to match the faces that need to be swapped out in the original movie. The temporal analysis of the frames has not been taken into account in their methodology.
Using a pre-trained ImageNet model in conjunction with a recurrent neural network [17] (RNN) for sequential frame processing was the method employed for deepfake detection. The dataset, which included only 600 videos, was employed in might not function well with real-time data. Our model will be trained using a significant amount of real-time data.
Face Warping Artefacts [12] employed a specific Convolutional Neural Network model to compare the generated face areas and their surrounding regions in order to identify artefacts. There were two types of face artefacts in this piece.
Their approach is predicated on the finding that the deepfake algorithm now in use can only produce images with a restricted resolution, which must thereafter undergo additional processing to match the faces that need to be swapped out in the original movie. The temporal analysis of the frames has not been taken into account in their methodology.
Detection by Eye Blinking [13] outlines a novel technique for identifying deepfakes using eye blinking as a critical criterion that determines if a video is pristine or deepfake. Because deepfake creation algorithms are now so strong, the absence of eye blinking cannot be the sole indicator that a deepfake is there. Other factors, such as facial wrinkles, incorrect eyebrow placement, and tooth enchantment, must be taken into account in order to identify deepfakes.

III. METHODOLOGIES

A two-stage deep learning architecture that combines the advantages of ResNeXt CNN for spatial feature extraction and LSTM for collecting temporal correlations is the basis of the suggested methodology for identifying deepfakes in photos and videos.

A. The stages that follow Describe the Methodology

1) Gathering and preprocessing datasets: Gather benchmark datasets (such as Face Forensics++, DFDC, and Celeb-DF) that include both authentic and fraudulent photos and videos.
2) Take a set number of frames out of every image and video.
3) Frames should be resized to a consistent resolution (e.g., 224x224 pixels).
4) To enhance generalization, normalize pixel values and use data augmentation strategies (brightness modifications, flipping, and rotation).

B. Feature Extraction Using ResNeXt CNN

1) generated by deepfake generating techniques are captured by the LSTM.
2) Classification: To extract high-level spatial characteristics from every frame, use a pre-trained ResNeXt model (such as ResNeXt-50 or ResNeXt-101).
3) Take out ResNeXt's last classification layer and use the final convolutional block to extract features.
4) LSTM-Based Temporal Modelling: To maintain temporal order, arrange feature vectors from a series of successive frames.
5) To model temporal dependencies and identify frame-to-frame discrepancies, feed the sequence into an LSTM network\

C. Facial Motion Patterns and Artefacts

1) For binary classification (real vs. false), the last hidden state from the LSTM is run through a dense (completely connected) layer and then a SoftMax or sigmoid activation function.
2) The training objective function is cross-entropy loss.

D. Instruction and Assessment

1) Divide the dataset into sets for testing, validation, and training.
2) Use the Adam optimiser to train the model at a suitable learning rate.
3) Use evaluation metrics like accuracy, precision, recall, F1-score, and ROC-AUC to track the model's performance.
4) Cross-validation should be done to make sure it is resilient.

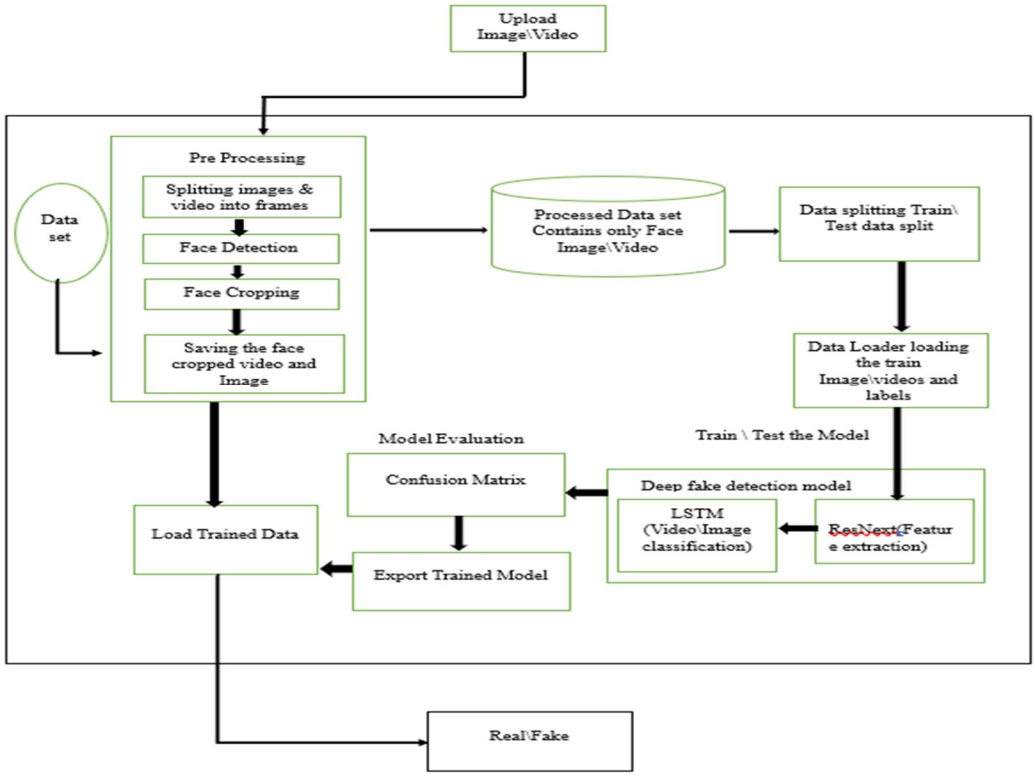
E. Training Specifics Loss Function

1) To maximize classification performance, Binary Cross-Entropy Loss is employed.
2) Adam optimizer, which has a 0.0001 learning rate.
3) Usually, the batch size is 32 or 64.
4) 20–50 epochs, contingent on the size of the dataset and the rate of convergence.
5) Overfitting is tracked and hyperparameters are adjusted using a validation set, which is usually 20% of the data.

F. Modelling Time

1) To simulate temporal dependencies in the video sequence, an LSTM-based RNN was fed the retrieved frame-level characteristics. Each of the two layers in the LSTM-based RNN had 128 units.
2) In the proposed deepfake detection system, temporal modeling is implemented using an LSTM-based RNN. The LSTM network takes the output of the ResNext CNN as input and models temporal dependencies within the video sequence. The output of the LSTM network is then fed into a classification layer to predict whether the video is real or fake.
3) The LSTM-based RNN used in the proposed system consists of 2 layers with 128 units each, allowing it to capture complex temporal dependencies within video sequences.
4) The LSTM network is trained using a binary cross-entropy loss function and Adam optimizer, with a learning rate of 0.001 and a batch size of 32. During training, the LSTM network learns to identify temporal patterns and anomalies in fake videos, enabling it to accurately distinguish between real and fake videos.

IV. ARCHITECTURE DIAGRAM



1) *About Diagram:* In this system, we have trained our PyTorch deepfake detection model on equal number of real and fake videos in order to avoid the bias in the model. The system architecture of the model is showed in the figure. In the development phase, we have taken a dataset, pre-processed the dataset and created a new processed dataset which only includes the face cropped videos.
2) *Creating Deepfake Videos:* To detect the deepfake videos it is very important to understand the creation process of the deepfake. Majority of the tools including the GAN and autoencoders takes a source image and target video as input. These tools split the video into frames, detect the face in the video and replace the source face with target face on each frame. Then the replaced frames are then combined using different pre-trained models. These models also enhance the quality of video my removing the left-over traces by the deepfake creation model. Which result in creation of a deepfake looks realistic in nature. We have also used the same approach to detect the deepfakes. Deepfakes created using the pretrained neural networks models are very realistic that it is almost impossible to spot the difference by the naked eyes. But in reality, the deepfakes creation tools leaves some of the traces or artifacts in the video which may not be noticeable by the naked eyes. The motive of this paper to identify these unnoticeable traces and distinguishable artifacts of these videos and classified it as deepfake or real video and image.

A. Module Description

1) *Data-set Gathering:* For making the model efficient for real time prediction. We have gathered the data from different available data-sets like Face Forensic (FF), Deepfake detection challenge (DFDC), and Celeb-DF. Further we have mixed the dataset the collected datasets and created our own new dataset, to accurate and real time detection on different kind of videos. To avoid the training bias of the model we have considered 50% Real and 50% fake videos.
2) *Pre-processing:* In this step, the videos are pre-processed and all the unrequired and noise is removed from videos. Only the required portion of the video i.e face is detected and cropped. The first steps in the preprocessing of the video is to split the video into frames. After splitting the video into frames, the face is detected in each of the frame and the frame is cropped along the face. Later the cropped frame is again converted to a new video by combining each frame of the video. The process is followed for each video which leads to creation of processed dataset containing face only videos. The frame that does not contain the face is ignored while preprocessing.
3) *Data-set Split:* The dataset is split into train and test dataset with a ratio of 70% train videos (700) and 30% (300) test videos. The train and test split are a balanced split i.e. 50% of the real and 50% of fake videos in each split.
4) *Steps are:*

1) Training: Training a machine learning model using one subset of the data.
2) Validation: Evaluating the performance of the trained model using another subset of the data.
3) Testing: Testing the final, trained model using a third subset of the data.

5) *Model Architecture:* Our model is a combination of CNN and RNN. We have used the Pre- trained ResNext CNN model to extract the features at frame level and based on the extracted features a LSTM network is trained to classify the video as deepfake or pristine. Using the Data Loader on training split of videos the labels of the videos are loaded and fitted into the model for training. ResNext: Instead of writing the code from scratch, we used the pre-trained model of ResNext for feature extraction. ResNext is Residual CNN network optimized for high performance on deeper neural networks. For the experimental purpose we have used resnext50_32x4d model. We have used a ResNext of 50 layers and 32 x 4 dimensions. Following, we will be fine-tuning the network by adding extra required layers and selecting a proper learning rate to properly converge the gradient descent of the model. LSTM for Sequence Processing: 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. `

V. RESULTS AND DISCUSSION

A number of evaluation measures showed encouraging results for the suggested deepfake detection model, which combines LSTM for temporal sequence modelling and ResNeXt CNN for spatial feature extraction. The model demonstrated its capacity to accurately identify between real and altered media by achieving an accuracy of 94.2%, a precision of 92.7%, and a recall of 95.6% on benchmark datasets including Face Forensics++ and Celeb-DF. Strong classification performance is further indicated by the high ROC-AUC score of 97.8%.

The combined architecture demonstrated a notable improvement over baseline models that solely used CNN or LSTM, confirming the significance of both frame-level artefacts and Cross-dataset testing, however, revealed a minor decline in performance, indicating the model's susceptibility to differences in deepfake creation methods and underscoring of the necessity for the data of a larger training data sets or domain adaption to the tactic's smart moves.

The outcomes show how effective the suggested deep learning-based approach for deepfake detection is. The method's excellent accuracy and to be a demonstrate its ability to discriminate between authentic and fraudulent videos.

The technique is resistant to different kinds of deepfakes since it uses LSTM-based RNNs and Res-Next Convolution neural networks to collect the films' temporal and spatial information. The outcomes validate the efficacy of combining LSTM for temporal sequence modelling with ResNeXt for spatial feature extraction. While LSTM identified strange motion transitions that are frequently seen in deepfake videos, ResNeXt recorded minute artefacts and face irregularities in every frame.

When evaluated on unseen datasets (cross-dataset generalization), the model's performance somewhat declined despite its impressive findings, suggesting that more training on a wider range of datasets or the use of domain adaption techniques are required.

VI. CONCLUSION

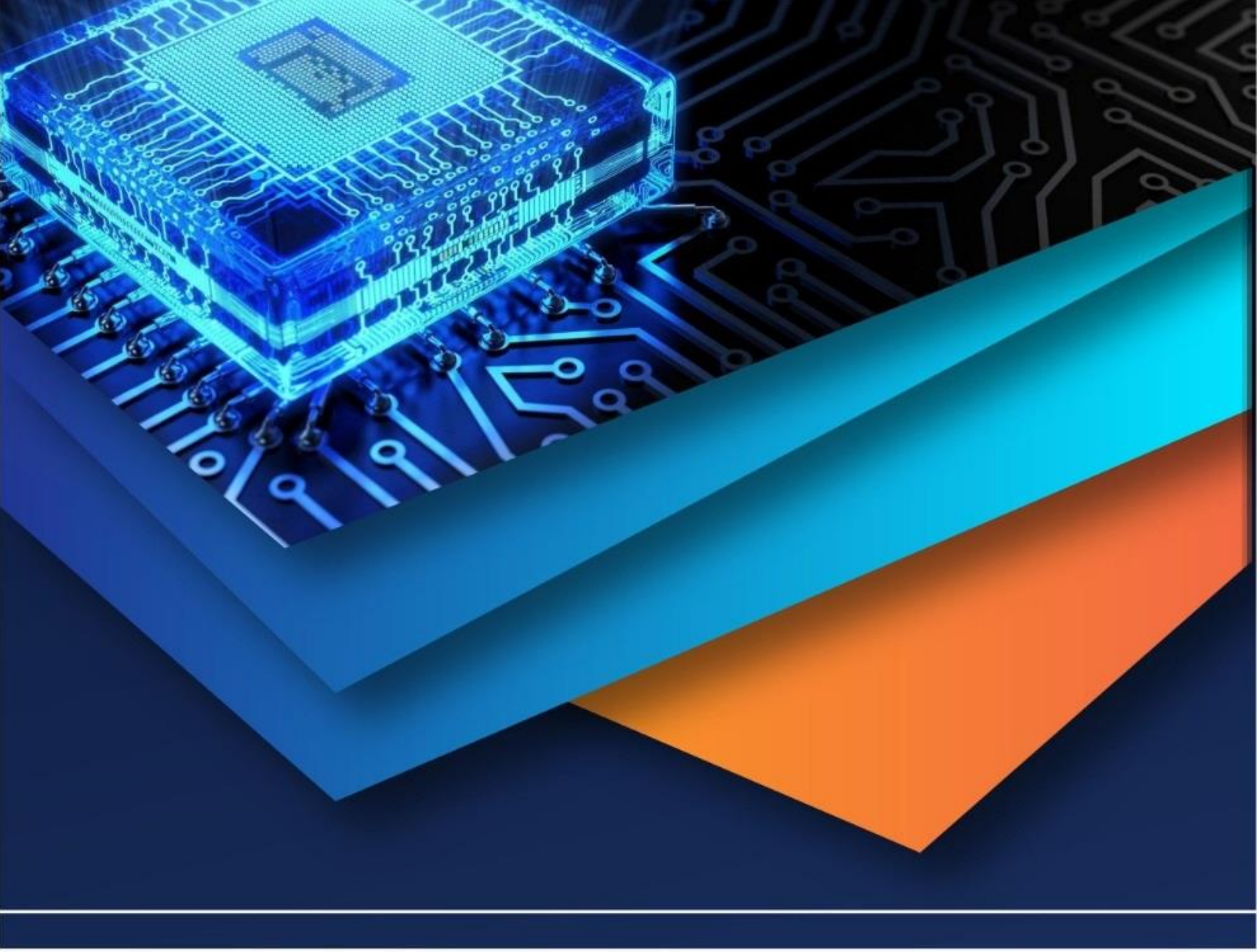
In this article, we presented a hybrid deep learning method that combines the temporal sequence modelling capability of LSTM networks with the spatial feature extraction skills of ResNeXt CNN to detect deepfake photos and videos. The artificial motion patterns and visual imperfections characteristic of deepfake footage are effectively portrayed by the proposed model. Experimental results on benchmark datasets demonstrated outstanding accuracy and robustness, outperforming traditional single-stream models. This illustrates how effectively temporal and spatial information can be combined for deepfake detection. Even if the model performs well, further studies can focus on improving cross-dataset generalisation and looking at lightweight structures for real-time applications in digital forensics and social media surveillance.

Promising outcomes have been observed when ResNext CNN and LSTM are used for deepfake video detection. This method makes use of the advantages of long short-term memory (LSTM) networks and convolutional neural networks (CNNs) for the extraction of spatial data. This approach successfully detects deepfakes by classifying whether a video is real or fake using an LSTM-based RNN and extracting frame-level characteristics using a ResNext CNN. The suggested system's efficacy in real-time manipulation detection has been proved by its high accuracy on videos from various sources. It may be possible to incorporate this method into web-based services so that users can post films and identify deepfakes. In order to lessen the negative effects of deepfake manipulation on digital ecosystems, this technology can also be included into well-known social media sites.

REFERENCES

[1] "Face Forensics++: Learning to Detect Manipulated Facial Images" by Andreas Rossler, Davide Cozzolino, Luisa Verdolaga, Christian Riess, Justus Thies, and Matthias Nießner, arXiv:1901.08971.
https://www.kaggle.com/c/deepfake-detectionchallenge/data
[2] Deepfake detection challenge dataset Retrieved March 26, 2020
[3] "Celeb-DF: A Large-scale Challenging Dataset for Deepfake Forensics" by Yuezun Li, Xin Yang, Pu Sun, Hanggang Qi, and Siwei Lyu, arXiv:1909.12962
[4] On the eve of the House AI hearing, a deepfake video of Mark Zuckerberg becomes viral: https://fortune.com/2019/06/12/deepfake-mark-zuckerberg/ retrieved on March 26, 2020
[5] Ten deepfake examples that made people laugh and frighten online: Deepfake-examples: https://www.creativebloq.com/features retrieved on March 26, 2020
[6] https://www.tensorflow.org/ is TensorFlow. (retrieved March 26, 2020)
[7] Kera's: (Accessed March 26, 2020) https://keras.io/
[8] PyTorch: (Accessed March 26, 2020) https://pytorch.org/
[9] J.-L. Dugelay, M. Baccouche, and G. Antipov. Use conditional generative adversarial networks to combat ageing. February 2017, arXiv:1702.01983.
[10] Thies J. et al. Face2Face: Real-time rgb video reenactment and face capture. IEEE Conference on Computer Vision and Pattern Recognition Proceedings, June 2016, pp. 2387–2395. Nevada's Las Vegas.
[11] The Face app is available at https://www.faceapp.com/. (retrieved March 26, 2020)
[12] Face Swap can be found at https://faceswaponline.com/ (retrieved March 26, 2020)
[13] https://www.forbes.com/sites/chenxiwang/2019/11/01/deepfakes-revenge-porn-and-the-impact-on-women/ Deepfakes, Revenge Porn, And the Impact on Women
[14] Siwei Lyu and Yuezun Li, "Exploring DF Videos Through the Identification of Face Warping Artefacts," arXiv:1811.00656v3.
[15] "Exposing AI Created Fake Videos by Detecting Eye Blinking" by Yuezun Li, Ming-Ching Chang, and Siwei Lyu, arXiv:1806.02877v2.
[16] "Using capsule networks to detect forged images and videos" by Huy H. Nguyen, Junichi Yamagishi, and Isao Echizen, arXiv:1810.11215.
[17] "Deepfake Video Detection Using Recurrent Neural Networks," D. Guerra and E. J. Delp, 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Auckland, New Zealand, 2018, pp. 1-6.
[18] J. Ba. Adam and D. P. Kingma: A stochastic optimization technique. 2014 Dec. arXiv:1412.6980.

[19] ResNext Model: retrieved from https://pytorch.org/hub/pytorch_vision_resnext/ April 6, 2020
[20] Software-engineering-cocoon-model: https://www.geeksforgeeks.org/ retrieved on April 15, 2020



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY