JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Deepfake Video Detection using Neural Networks

[1]Dheeraj Yadav, [2]Pravin Modi, [3]Nikhil Yadav, [4]Avanish Kumar

[1]Department of Computer Engineering,
[1]Lokmanya Tilak College of Engineering, Navi Mumbai, India

Abstract : In recent months, free deep learning-based software tools has facilitated the creation of credible face exchanges in videos that leave few traces of manipulation, in what they are known as "DeepFake"(DF) videos. Manipulations of digital videos have been demonstrated for several decades through the good use of visual effects, recent advances in deep learning have led to a drastic 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 DF).Creating the DF using the Artificially intelligent tools are simple task. But, when it comes to detection of these DF, it is major challenge. Because training the algorithm to spot the DF is not simple. We have taken a step forward in detecting the DF using Convolutional Neural Network and Recurrent neural Network. System uses a convolutional Neural network (CNN) to extract features at the frame level. These features are used to train a recurrent neural network (RNN) which learns to classify if a video has been subject to manipulation or not and able to detect the temporal inconsistencies between frames introduced by the DF creation tools. Expected result against a large set of fake videos collected from standard data set. We show how our system can be competitive result in this task results in using a simple architecture.

IndexTerms **-Deepfake Video Detection, convolutional Neural network (CNN), recurrent neural network**

I. Introduction

The rapid advancement of smartphone camera technology and widespread internet connectivity has significantly increased the creation and distribution of digital videos. Social media platforms and media-sharing portals have made it easier than ever to share content globally. Simultaneously, the rise in computational power has fueled the growth of deep learning techniques, enabling complex tasks that were once considered impossible. Among these advancements, Generative Adversarial Networks (GANs) have led to the emergence of highly realistic synthetic media, commonly known as DeepFakes (DFs).

DeepFake technology allows for the seamless manipulation of video and audio, making it possible to replace faces in videos with astonishing accuracy. While this innovation has promising applications in entertainment and education, it also poses significant challenges, particularly in misinformation, identity fraud, and cyber threats. The rapid spread of DeepFake content on social media platforms has contributed to the dissemination of false information, leading to serious societal concerns. These manipulated videos can be used for political propaganda, defamation, and various other malicious purposes, making their detection a crucial research challenge.

To address these concerns, DeepFake detection techniques have become essential. This paper proposes a novel deep learning-based approach to accurately distinguish AI-generated fake videos from real ones. Our method focuses on identifying the subtle artifacts left by GAN-generated content. Due to computational constraints, DeepFake models typically synthesize face images at a fixed resolution and then apply affine transformations to match the target video. This process introduces inconsistencies, such as resolution mismatches and unnatural warping effects, which can be leveraged for detection.

Our approach employs a ResNext Convolutional Neural Network (CNN) to extract spatial features from video frames and a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) to capture temporal inconsistencies across frames. By analyzing these artifacts, our model effectively differentiates between DeepFake videos and authentic content. The proposed method enhances the reliability of DeepFake detection and contributes to the growing field of AI-driven cybersecurity and digital forensics.

2. Literature Survey :

Deepfake Detection Methods and Approaches

1. **Exposing Deepfake Videos by Detecting Face Warping Artifacts [1]**
   - Uses a Convolutional Neural Network (CNN) to detect artifacts in generated face areas.
   - Identifies discrepancies between generated face areas and their surrounding regions.
   - Observes that deepfake algorithms generate images of limited resolution, which are then transformed to fit the source video.

2. **Exposing AI-Created Fake Videos by Detecting Eye Blinking [2]**
   - Detects fake videos by analyzing eye-blinking patterns.
   - Eye blinking is a physiological trait often missing in deepfake videos.
   - Evaluated on eye-blinking detection benchmarks, showing promising results.
   - Limitation: Only considers the lack of blinking as a detection clue.
   - Suggests incorporating additional parameters like teeth enhancement, facial wrinkles, etc.

3. **Using Capsule Networks for Deepfake Detection [3]**
   - Employs capsule networks to detect manipulated images and videos.
   - Effective for replay attack detection and computer-generated video detection.
   - Uses random noise in training, which may reduce real-time accuracy.
   - Proposed improvement: Training on noiseless and real-time datasets.

4. **Detection of Synthetic Portrait Videos Using Biological Signals [5]**
   - Extracts biological signals from facial regions in real and fake videos.
   - Uses transformations to analyze spatial coherence and temporal consistency.
   - Captures signal characteristics in feature sets and PPG (Photoplethysmography) maps.
   - Trains a probabilistic SVM and CNN for classification.

5. **Fake Catcher – A High-Accuracy Deepfake Detection System**
   - Detects fake videos independently of generator type, content, resolution, and quality.
   - Uses biological signals to determine authenticity.
   - Limitation: Lack of a discriminator may lead to loss of biological signals.
   - Requires formulating a differentiable loss function aligned with signal processing steps.

3. Methodology:

There are many tools available for creating the DF, but for DF detection there is hardly any tool available. Our approach for detecting the DF will be great contribution in avoiding the percolation of the DF 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 automatic DF detections.
Even big application like WhatsApp, Facebook can integrate this project with their application for easy pre detection of DF before sending to another user. One of the important objective is to evaluate its performance and acceptability in terms of security, user-friendliness, accuracy and reliability. Our method is focusing on detecting all types of DF like replacement DF, retrenchment DF and interpersonal DF. figure.1 represents the simple system architecture of the proposed system: -

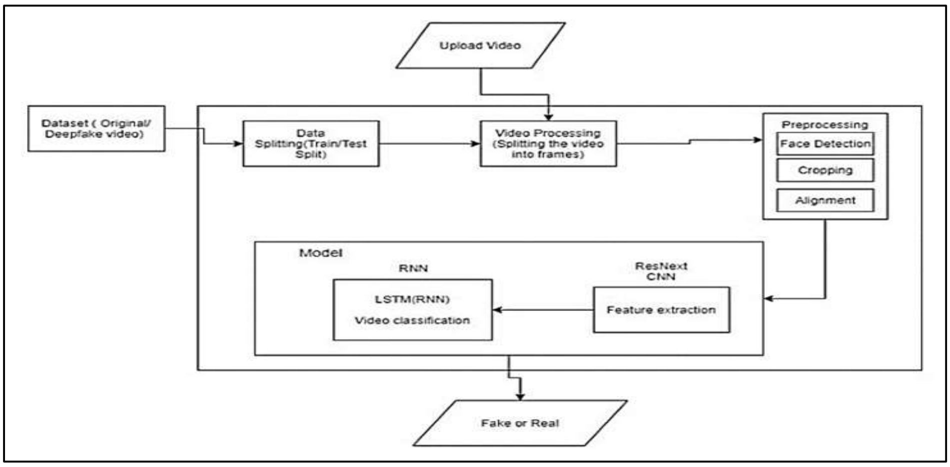

Fig. 1: System Architecture

A. *Dataset:*

We are using a mixed dataset which consists of equal amount of videos from different dataset sources like YouTube, FaceForensics++[14], Deep fake detection challenge dataset[13]. Our newly prepared dataset contains 50% of the original video and 50% of the manipulated deepfake videos. The dataset is split into 70% train and 30% test set.

B. *Preprocessing:*

Dataset preprocessing includes the splitting the video into frames. Followed by the face detection and cropping the frame with detected face. To maintain the uniformity in the number of frames the mean of the dataset video is calculated and the new processed face cropped dataset is created containing the frames equal to the mean. The frames that doesn't have faces in it are ignored during preprocessing.

As processing the 10 second video at 30 frames per second i.e total 300 frames will require a lot of computational power. So for experimental purpose we are proposing to used only first 100 frames for training the model.

C. *Model:*

The model consists of resnext50_32x4d followed by one LSTM layer. The Data Loader loads the preprocessed face cropped videos and split the videos into train and test set. Further the frames from the processed videos are passed to the model for training and testing in mini batches.

D. *ResNext CNN for Feature Extraction*

Instead of writing the rewriting the classifier, we are proposing to use the ResNext CNN classifier for extracting the features and accurately detecting the frame level features. 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. The 2048-dimensional feature vectors after the last pooling layers are then used as the sequential LSTM input.

E. *LSTM for Sequence Processing*

Let us assume a sequence of ResNext CNN feature vectors of input frames as input and a 2-node neural network with the probabilities of the sequence being part of a deep fake video or an untampered video. The key challenge that we need to address is the de- sign of a model to recursively process a sequence in a meaningful manner. For this problem, we are proposing to the use of a 2048 LSTM unit 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

F. Predict:

A new video is passed to the trained model for prediction. A new video is also preprocessed to bring in the format of the trained model. The video is split into frames followed by face cropping and instead of storing the video into local storage the cropped frames are directly passed to the trained model for detection.

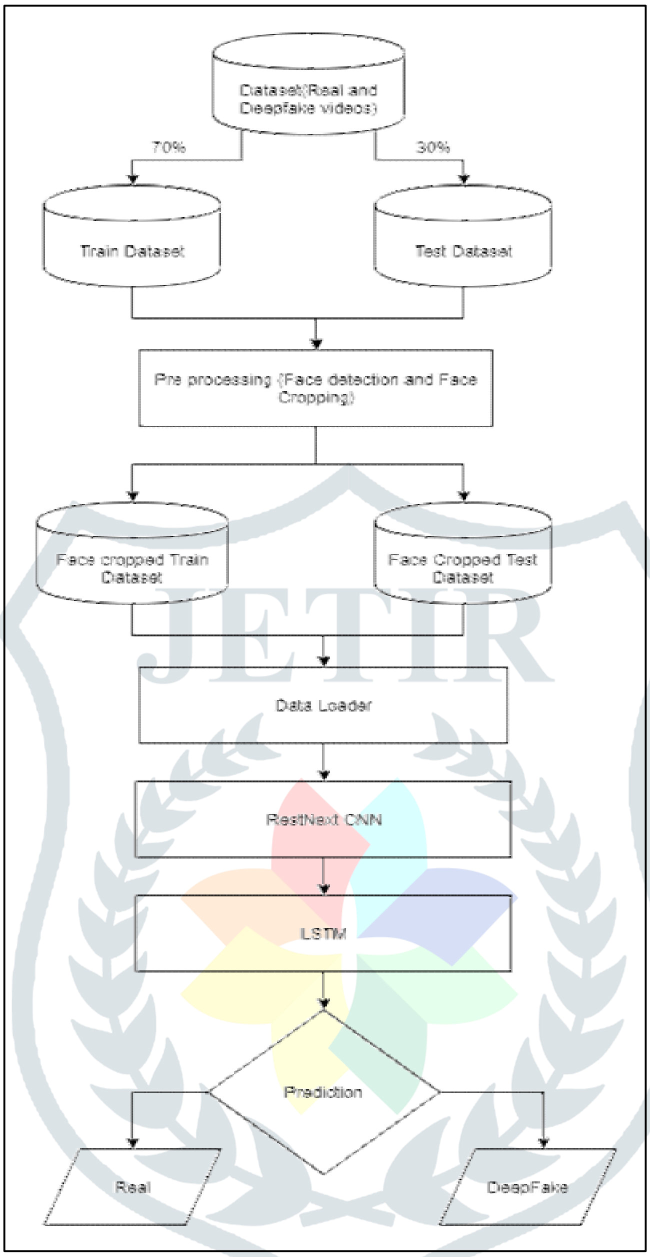
Dataset(Real and Deepfake videos)
70%
30%
Train Dataset
Test Dataset
Pre processing (Face detection and Face Cropping)
Face cropped Train Dataset
Face Cropped Test Dataset
Data Loader
RestNext CNN
LSTM
Prediction
Real
DeepFake

Fig. 2: Training Flow

4. RESULT

The output of the model is going to be whether the video is deepfake or a real video along with the confidence of the model. One example is shown in the figure 3.

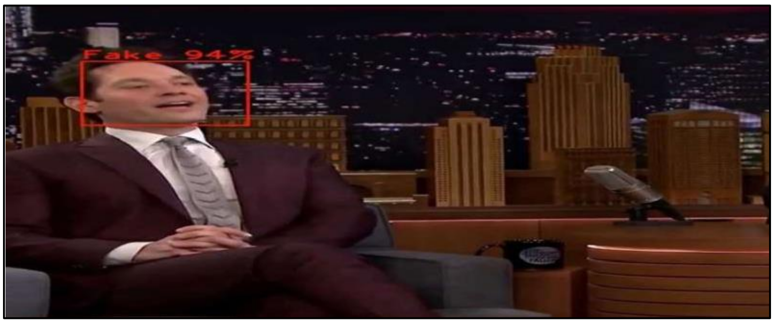

Fig. 3: Expected Results

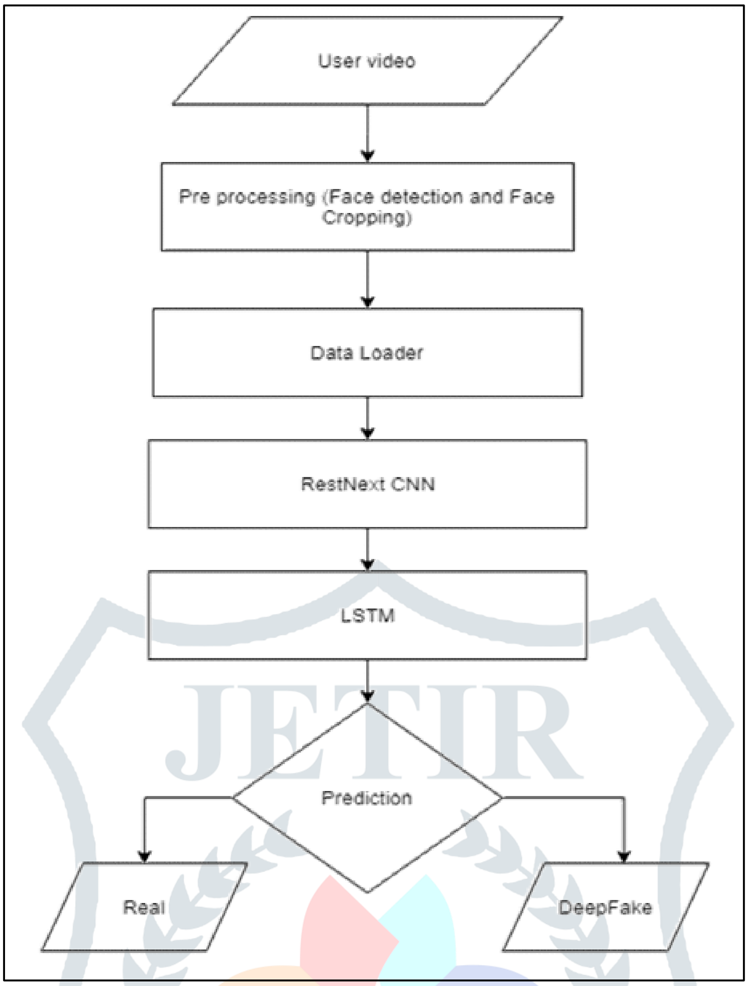

Fig. 4: Prediction flow

5. Conclusion

We presented a neural network-based approach to classify the video as deep fake or real, along with the confidence of proposed model. The proposed method is inspired by the way the deep fakes are created by the GANs with the help ofAutoencoders. Our method does the frame level detection using ResNext CNN and video classification using RNN along with LSTM. The proposed method is capable of detecting the video as a deep fake or real based on the listed parameters in paper. We believe that, it will provide a very high accuracy on real time data.

6. Limitations

Our method has not considered the audio. That’s why our method will not be able to detect the audio deep fake. But we are proposing to achieve the detection of the audio deep fakes in the future.

7. References

[1] Yuezun Li, Siwei Lyu, “ExposingDF Videos By Detecting Face Warping Artifacts,” in arXiv:1811.00656v3.
[2] Yuezun Li, Ming-Ching Chang and Siwei Lyu “Exposing AI Created Fake Videos by Detecting Eye Blinking” in arxiv.
[3] Huy H. Nguyen , Junichi Yamagishi, and Isao Echizen “ Using capsule networks to detect forged images and videos ”.
[4] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari and Weipeng Xu “Deep Video Portraits” in arXiv:1901.02212v2.
[5] Umur Aybars Ciftci, ˙Ilke Demir, Lijun Yin “Detection of Synthetic Portrait Videos using Biological Signals” in arXiv:1901.02212v2.
[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In NIPS, 2014.
[7] David G¨uera and Edward J Delp. Deepfake video detection using recurrent neural networks. In AVSS, 2018.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.

[9] An Overview of ResNet and its Variants : https://towardsdatascience.com/an-overview-of-resnet- and-its-variants-5281e2f56035

[10] Long Short-Term Memory: From Zero to Hero with Pytorch: https://blog.floydhub.com/long-short-term-memory-from-zero-to-hero-with-pytorch/

[11] Sequence Models And LSTM Networks https://pytorch.org/tutorials/beginner/nlp/sequence_mod els_tutorial.html

[12] https://discuss.pytorch.org/t/confused-about-the-image- preprocessing-in-classification/3965

[13] https://www.kaggle.com/c/deepfake-detection- challenge/data

[14] https://github.com/ondyari/FaceForensics

[15] Y. Qian et al. Recurrent color constancy. Proceedings of the IEEE International Conference on Computer Vision, pages 5459–5467, Oct. 2017. Venice, Italy.

[16] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros. Image-to- image translation with conditional adversarial networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5967–5976, July 2017. Honolulu, HI.

[17] R. Raghavendra, Kiran B. Raja, Sushma Venkatesh, and Christoph Busch, "Transferable deep-CNN features for detecting digital and print-scanned morphed face images," in CVPRW. IEEE, 2017.

[18] Tiago de Freitas Pereira, Andr´e Anjos, Jos´e Mario De Martino, and S´ebastien Marcel, "Can face anti spoofing countermeasures work in a real world scenario?,"in ICB. IEEE, 2013.

[19] Nicolas Rahmouni, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, "Distinguishing computer graphics from natural images using convolution neural networks," in WIFS. IEEE, 2017.

[20] F. Song, X. Tan, X. Liu, and S. Chen, "Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients," Pattern Recognition, vol. 47, no. 9, pp. 2825–2838, 2014.. E. King, "Dlib-ml: A machine learning toolkit," JMLR, vol. 10, pp. 1755–1758, 2009.