Combining Deep Learning and Super-Resolution Algorithms for Deep Fake Detection

Nikita S. Ivanov[1], Anton V. Arzhskov[2], Vitaliy G. Ivanenko[3]
Department of Computer Systems and Technologies
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute)
Moscow, Russian Federation
[1]ivanov.affairs@gmail.com, [2]zdj22@yandex.ru, [3]VGIvanenko@mephi.ru

***Abstract*—Deep Fake is a technique for human image synthesis based on artificial intelligence. In this article is explored the problem of Deep Fake Video content and its detection. Has been gathered information about previous attempts, analyzed methods used by different researches and considered their actuality right now. Basing on results of the discovery was designed strategy to expose Deep Fake videos that combines previous detection methods with super-resolution algorithms. Results of the research were compared with expected, so recommendations and possible way of continuing developments were given.**

Keywords—deep fake detection; deep learning; neural networks; super-resolution algorithms.

I. Introduction

In last year's Deep Fake (Fig. 1) inflamed society around the world because of the new app "Fake app" that made Deep Fake technology accessible for common users: every user got a powerful instrument to make their own new fakes. Deep Fake provides possibilities for malicious hoaxes which can violate official rules as well as ethical norms.

Fig. 1.Fake(left) and Real(right) image example. (Examples from UADFV dataset)

At the same time, Deep Fakes detection became one of popular research issues. Many attempts have been done in order to find a solution to this problem. Frequently, solution methods have used visible artifacts, that are common among most of Deep Fakes. The most successful methods are based on eye blinking [1], mismatched color profiles [2] and face warping artifacts [3]. Such artifacts provide good accuracy on big part of deep fakes, especially on old ones. On the other side, issue is considered as more complex, requiring other correlations besides visual ones. Due to this, we have attempts that are directly classifying faked content using actual algorithms of machine and deep learning [4]. Another look to this problem gave the different approach to Deep Fake detection. Method was designed to expose Deep Fakes basing on mismatch between directions of different face regions [5]. It gave good accuracy on more accurate fakes, but still have problems with low resolution video fakes.

II. Research Materials And Methods

Basing on the results of previous research we suggest the new method for detecting Deep Faked video content. Our method is inherently an alliance of 2 methods: Exposing Deep Fakes using inconsistent head poses [5] and detecting Deep Fake pictures using CNN Resnet50 [6] model. Pipeline of our system (Fig. 2) consists of 4 blocks: Dataset preprocessing, ResNet Classification, Inconsistent Head Pose estimator and Arbitrage (decision maker). At first, original videos fed to Face Recognition module. There they are separated for frames; face location is estimated on each frame and 68 face landmarks (Fig. 3) are estimated for each frame. Then all preprocessed frames are going through 2 classifications. In addition to ResNet, the super-resolution algorithm was applied to solve this problem. It aims to increase the accuracy of predictions on low-resolution videos: the quality of Deep Fakes often artificially decreased in order to hide artifacts and make pictures closer to the real one. Results from both classifications are transferred to Decision maker that is announcing the final judgement.

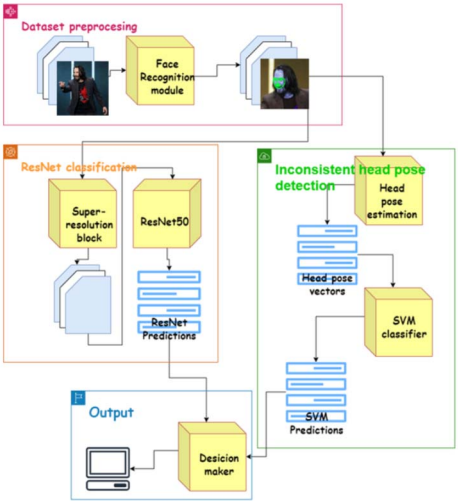

Fig. 2.Detection system pipeline

978-1-7281-5761-0/20/$31.00 ©2020 IEEE

A. *ResNet classifier.*

The hole research could be classified for several sub-objectives. Firstly, single image face detection methods for video content were applied: video is considered as a list of frames. For each frame we applied face detection methods realized in Dlib software library [7] and face recognition python module [8].

Basing on these landmarks the face rectangle is built which is used to crop all frames. Cropped frames, containing only face regions, are further used for training and evaluating classifying network.

In our research we implemented Resnet50v2 model which was finetuned for binary classification purposes. Weights of pretrained on ImageNet dataset were used as a base of our learning process. To finetune model instructions from "Exposing DeepFake Videos By Detecting Face Warping Artifacts." were followed (we took learning rate for 0.001, momentum equal to 0.9, used Cross Entropy loss function and set SGD optimizer), although the expected results of near 95 accuracy weren't achieved [3]. So, it was decided to increase the learning rate of the model from 0.001 to 0.01, owing to that the accuracy about 94.9% on evaluation dataset was achieved. The model was trained for 20 epochs, evaluated after each epoch and its parameters were saved. The best were picked for further research.

Fig. 3. 68 face landmarks got using Dlib package

B. *Inconsistent Head Pose estimator.*

Another big part of our study was reproducing method used in Li Yuezun, Lyu Siwei research [5]. The main idea consists in difference of estimated face direction vectors: the one based on outer landmarks and the one based on inner landmarks. During mask reconstruction Deep Fake algorithms inevitably make invisible difference in face directions between outer and inner parts. Thanks to that vector analyze gives program an opportunity to expose Deep Fakes even when sophisticated eye can't recognize it. For head pose estimation one may need to find 2 vectors (rotation vector and translation vector), which relates world coordinates of facial landmarks and their locations on the image. Together these 2 vectors let the one to build head pose vector. In few words, the following objective can be formulated as optimization(minimalization) problem for function:

$$\sum_{i=1}^{n} \left\| s\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} - \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \left(R\begin{bmatrix} U_i \\ V_i \\ W_i \end{bmatrix} + \vec{t} \right) \right\|^2 \quad (1)$$

Where x_i, y_i – image coordinates of landmark, f_x, f_y – focal lengths in x and y directions, c_x, c_y – coordinates of optical center, s – scaling factor (basically set to 1), U_i, V_i, W_i – world 3D coordinates of face landmark, R – estimated camera pose, which could be easily transform to head pose equal to R_T, t - estimated translation vector. The goal is to estimate R and t basing on 2D and 3D world coordinates of face landmarks. World landmarks coordinates were gathered using standard face model, which was uploaded in Autodesk Fusion 360. Landmarks were manually set and then their coordinates were written. The Equation 1 is successfully solved using OpenCV2 [9] build-in function SolvePnP. Following foregoing research, we used landmarks $18-36$, 49, 55 for estimating v_c – vector of inner parts of face and landmarks $1-36$, 49, 55 for detecting v_a – vector for outer parts of head. To make a prediction we used implemented in Scikit-learn [10] SVM classifier to conduct binary classification. The model was fed both v_c and v_a vectors and used default arguments.

C. *Super-resolution preprosessing.*

The last part of our research was applying super-resolution algorithms, so it was decided to implement Fast Super-resolution CNN model [11], which would be able to upsample data from single image [12]. CNN consists of 5 convolutional layers and a deconvolutional layer. For each layer was set a PReLU activation function, which provides better and more stable performance in comparison with ReLU ones. The model was implemented using PyTorch functions [13].

III. RESULTS

As the base for our research UADFV [5] dataset was chosen. This dataset contains 49 real and 49 fake videos of medium quality: Deep Fakes in this dataset could be recognized by human, but it still makes interest for us as a good point to start from.

To estimate the effect of combined methods for Deep Fake detection, first, several experiments were conducted with separated parts of our method. Resnet50v2 model was estimated alone on UADFV dataset and showed the highest result of 94.9% accuracy (Fig. 5).

The same experiment was conducted for second method using estimated head direction vectors. Due to unfound reason estimation haven't shown any good results, the accuracy of our classifier hasn't moved upper than 50.1%. We suggest this failure was caused by mistakes made while choosing world coordinates: the gathering from standard face model may lead to false estimation of R and t vectors.

To train FSRCNN were used 7000 images from CelebA dataset [14]. In order to conduct training one may need dataset of LR images, so it was decided to take CelebA dataset images (Fig. 4), resize and downsample them using Scikit-Image module [15]. Artificial downsampling is very applicable to real fakes, because Deep downsampling is very applicable to real fakes, because Deep Fakes usually processed through similar

algorithms. FSRCNN was trained on 80% of data and tested on other part.

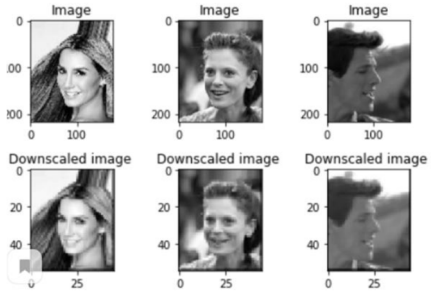

Fig. 4.CelebA dataset pictures and their downsamples made using Scikit-Image

At last, all components were gathered in a pipeline as showed on Figure 2. Whole system was evaluated on UADFV dataset and showed the final accuracy about 95.5%.

IV. Discussion And Conclusions

Our research faced several obstacles during its implementation. The accuracy of Resnet50v2 model alone and in system doesn't differs in this experiment. Obvious, the big part in this result played failure with head pose estimation and further classification (Fig. 6). So, this experiment can be considered as consisting only of 2 elements: Resnet classification and preprocessing using FSRCNN model. The results show that super-resolutioning before predicting doesn't make evident effect on the prediction. This result was tested only on videos of UADFV dataset, so there could be a reason to conduct some other testes on content with lower or higher resolution.

Fig. 5. Visualization of wrong head pose detection

The research could be further developed in several ways. At first, research could be repeated applying another popular method for Deep Fake detection such as one using RNN models. At the same time, for ResNet50v2 model could be replaced another ResNet model, e.g. ResNet20 and ResNet101. Experiments can be done with other methods of super-resolutioning and other models, such as SRCNN and EvoNet-based models [16].

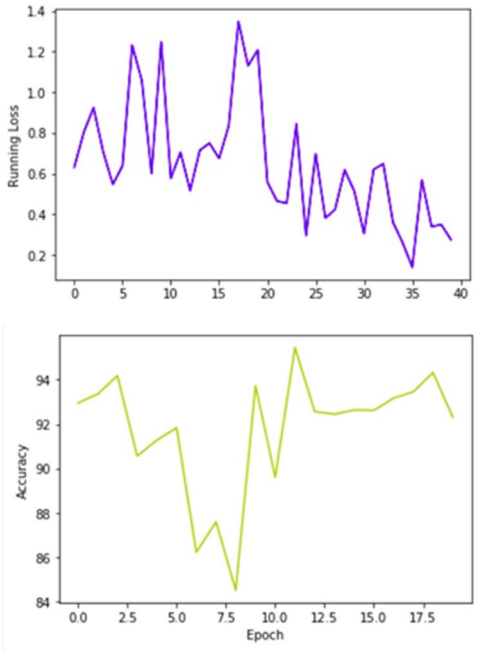

Fig. 6. Accuracy and Loss Plots for ResNet50v2 model trained on UADFV dataset

References

[1] Y. Li, M. C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI created fake videos by detecting eye blinking," in 10th IEEE International Workshop on Information Forensics and Security, WIFS 2018, 2019.

[2] H. Li, B. Li, S. Tan, and J. Huang, "Detection of Deep Network Generated Images Using Disparities in Color Components," Aug. 2018.

[3] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts."

[4] D. Guera and E. J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks," in Proceedings of AVSS 2018 - 2018 15th IEEE International Conference on Advanced Video and Signal-Based Surveillance, 2019.

[5] X. Yang, Y. Li, and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2019.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, vol. 2016-Decem, pp. 770–778.

[7] D. E. King, "Dlib-ml: A machine learning toolkit," J. Mach. Learn. Res., 2009.

[8] Adam Geitgey, "Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning," Medium, 2016.

[9] G. Bradski, "The OpenCV Library," Dr Dobbs J. Softw. Tools, 2000.

[10] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., 2011.

[11] C. Dong, C. C. Loy, and X. Tang, "Accelerating the Super-Resolution Convolutional Neural Network," Aug. 2016.

[12] M. Kawulok, P. Benecki, S. Piechaczek, K. Hrynczenko, D. Kostrzewa, and J. Nalepa, "Deep Learning for Multiple-Image Super-Resolution."

[13] A. Paszke et al., "Automatic differentiation in PyTorch."

[14] Z. Liu, P. Luo, X. Wang, and X. Tang, "Large-scale celebfaces attributes (celeba) dataset," Retrieved August, 2018.

[15] S. Van Der Walt et al., "Scikit-image: Image processing in python," PeerJ, vol. 2014, no. 1, 2014.

[16] M. Kawulok, P. Benecki, S. Piechaczek, K. Hrynczenko, D. Kostrzewa, and J. Nalepa, "Deep Learning for Multiple-Image Super-Resolution."