Methods of Deepfake Detection Based on Machine Learning

Artem A. Maksutov[1], Viacheslav O. Morozov, Aleksander A. Lavrenov, Alexander S. Smirnov
Department of Computer Systems and Technology
National Research Nuclear University "MEPhI"
Moscow, Russian Federation
[1]aamaksutov@mephi.ru

Nowadays, people faced an emerging problem of AI-synthesized face swapping videos, widely known as the DeepFakes. This kind of videos can be created to cause threats to privacy, fraudulence and so on. Sometimes good quality DeepFake videos recognition could be hard to distinguish with people eyes. That's why researchers need to develop algorithms to detect them. In this work, we present overview of indicators that can tell us about the fact that face swapping algorithms were used on photos. Main purpose of this paper is to find algorithm or technology that can decide whether photo was changed with DeepFake technology or not with good accuracy.

Keywords—deep learning; DeepFake detection; neural networks, face swapping indicators

I. INTRODUCTION

The idea of substituting face on photo is not so new as we can suggest. We can find examples of such photos made in 19th century. As an illustration, photo where the U.S.A. President Lincoln's head was swapped with politician John Calhoun's head was produced in mid-19th century. It was made by human hand. However, when the idea of neural networks became popular and humanity improved its computational skills, people began to use this technology in their everyday life. Nowadays we can download and run such programs - they can help us to get experience by experimenting without having a Ph.D. in math, computer theory, psychology, and more.

Today, none of us will be surprised by apps like FaceApp that have an opportunity to swap faces with good quality and make us funny. That became so, thanks to the work of one enthusiast: he brought together confusing and fragmentary code of groundbreaking face swapping technology and made it work on his own PC. When it became public domain, people immediately began to use it to create inappropriate content. For example, fake celebrity pornographic videos or revenge porn. But such videos and photos are more like baby pranks and no more. Great threat of Deepfake content came later. When it became hard to recognize by people eyes whether video was changed or not, real threats – fraudulent videos have come.

Nowadays, there are three most dangerous ways of using face swapping algorithms: face-swap, in which the face in a video is automatically replaced with another person's face; lip-sync, in which only the mouth region of face is changed and people on video are made to say something that they had never said (for example, a video where former USA President Obama is altered to say things like "President Trump is a total and complete dip-****."); and the most dangerous – puppet-master, in which target person's face is animated by person, sitting in front of camera.

Fig. 1. Example of good quality fake photo with face swapping algorithm used

When it all came to that, researchers decided to find ways of deepfake detection to protect people from this tremendous danger. That was the time, when the "competition" between offensive deepfakes and defensive methods of detection started. This is one reason why first datasets, containing AI-generated fake videos appeared.

We should notice that this "competition" made malefactors to upgrade their face swapping techniques to create more and more realistic fake videos. That's why pessimistic opinions appeared in researchers environment. Some of them started to think that it all looks like the antibiotics problem: viruses only become stronger, because of uncontrollable distribution of antibiotics usage. Like the battle with hydra: if you cut off one head, two will grow in its place.

But we think that it is only partially true. New effective methods of deepfake detection can stop inappropriate usage of fake videos for some time that we'll have to think about new defense types.

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

In this paper we present algorithm which can decide whether photo was changed with DeepFake face swapping technology or not with high accuracy.

II. Creating Deepfake Videos

There are 2 common ways of creating Deepfake videos: autoencoders and GANs. Both of them are being applied now. And we should talk about both of them if we want to create really good detecting algorithm.

III. Autoencoders

It is one of well-known deep learning techniques. Autoencoder insight is pair of encoder and decoder functions with shared weights, so 2 functions are training together. Usually, we talk about autoencoders in cases of dimensional reduction and generative models learning. But autoencoders can be used for taking compressed representations of images to outdo existing image compressing standards. So we can use this representations (called autoencoder's latent vectors) from first encoder with decoder part of second autoencoder and get resulting mix of pictures - face swapped photo, for example. Open-source repositories using this technique are: Faceswap, DFaker, DeepFakeLab and etc.

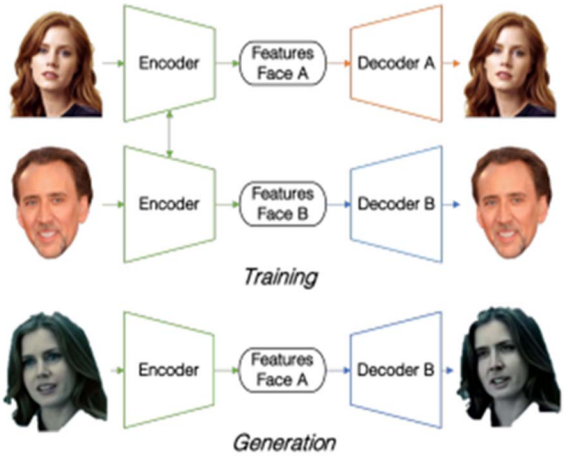

Fig. 2. Two networks sharing the same encoder, yet using two different decoders (top). When we want to do a new faceswapp, we encode the input face and decode it using the target face decoder (bottom)

IV. GANs

One of the most hard to train and use on computational hand deep learning technique is GAN. There we have 2 neural networks – generator and discriminator (judge, classifier, etc.). Generator net looks similar to autoencoder net, but we can achieve better results because discriminator net is rejecting some bad examples. So GANs technique of deepfake creation assumes that generator should fool discriminator (another machine) that makes such fakes more similar to real videos and also makes them harder recognizable by people's eyes. Some of the open-source projects are using this technique, for example, Faceswap-GAN.

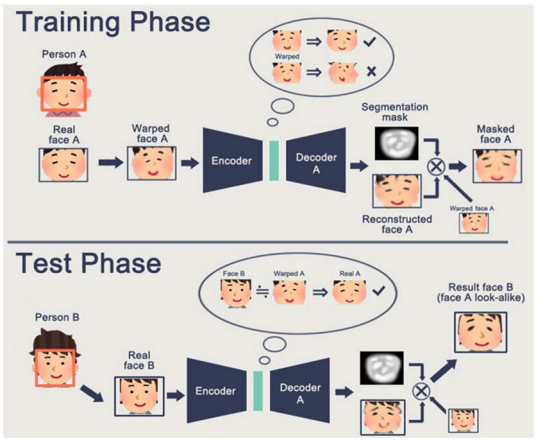

Fig. 3. Training GAN example

TABLE I. Summary of Notable Deepfake Tools

Tools	Links	Key features
Faceswap	https://github.com/deepfakes/faceswap	- Using two encoder-decoder pairs. - Parameters of the encoder are shared.
Faceswap-GAN	https://github.com/shaoanlu/faceswap-GAN	Adversarial loss and perceptual loss (VGGface) are added to the auto-encoder architecture.
DFaker	https://github.com/dfaker/df	- DSSIM loss function is used to reconstruct face. - Implemented based on Keras library.
DeepFaceLab	https://github.com/iperov/DeepFaceLab	- Expand from the Faceswap model with new models, e.g. H64, H128, LIAEF128, SAE. - Support multiple face extraction modes, e.g. S3FD, MTCNN, dlib, or manual.
DeepFake-tf	https://github.com/StromWine/DeepFake-tf	Similar to DFaker but implemented based on tensorflow.

V. Usual indicators of Deepfake videos

When we talk about Deepfake detection, obvious things that can tell us about video/photo "fakeness" can be finded. There are some usual indicators and we should check them out.

- Too smooth skin, lack of skin details – this indicators are consequence of one problem in DeepFake algorithms: low resolution of synthesized faces. But sometimes detection can be very hard, especially because of makeup on one of two faces. Original DeepFake algorithm generate faces of 64x64 pixels so we usually need to resize them. Now, some of the algorithms can produce 128x128 or even 256x256 faces but even such sizes can be not enough for good DeepFake video.
- Color mismatch between the synthesized face and the original face - this indicator can be used in human DeepFake recognition, but sometimes such mismatches can be very tricky to detect by eyes. But not for good program.
- Visible parts of original face or temporal flickering - when face swapping algorithm got improper choice of the face region we can see artifacts of the original face or even whole original face flickering. May be it is just one frame of the whole one-hour video. But we should check this frame more precisely.
- Head position – this indicator can appear due to the problem, described above.
- Artifacts on small moving parts – due to resolution limits, DeepFake algorithm cannot produce small moving parts with good quality. That's why we can sometimes see artifacts on hairs, eyebrows, eyelashes or some small skin defects.
- Eye blinking rate – indicator that was very useful in the very beginning of the face swapping algorithms popularity. Due to small datasets of photos and very small amount of eye-closed pictures there DeepFake couldn't produce an eye-blinking face and so blinking rate reduces. Now new versions of algorithms solved such problem, so it's not very helpful anymore.
- Face warping artifacts – one of the best indicators of fake videos, generated by algorithms with low resolution face output (64x64 or 128x128). After such small picture synthesized it should be transformed affinely. So some artifacts can be seen clearly. As another plus of such indicator is that we don't need deepfake datasets to train model. We can just use face detection algorithms and make some affine transformations to it. Face warping artifacts indicator may be the best choice right now, but when new face swapping algorithm and technologies appear and higher quality face pictures will be synthesized it can become useless.
- Person's patterns of behavior – can be useful, when we talk about the puppet-master and lip-sync techniques of Deepfakes. We can take usual person's behavior, get some patterns of it and try to identify similarity of usual and video behavior. It is, may be, the best indicator of fake videos, but it's very hard to use such indicator on photos and it can detect only fakes with person whose behavior patterns were taken.

VI. Model Experiments

As for model we use DenseNet169 with face warping artifacts indicator. It should work correctly with present algorithms of Deepfake so we decided to try it. Another reason of using this kind of model is that we don't actually need to train any face swapping algoithms to collect training and validation dataset. We just needed to collect photos of people from the internet. For negative examples we used this photos and added some noise to them. We tried Gaussian blur, exponential blur and Rayleigh blur. That is a quite simple idea, but when it comes to testing we don't usually get Gaussian noise after affining source picture. So we have to try different variants.

For extracting faces from the picture we use dlib package. After that we use random affine transformations on randomly resized pictures. Then we add random specific blur. Finally, face pictures are resized back and the picture is made whole again.

To evaluate our model we use Celeb-DF dataset, which is one of the newest datasets of Deepfake videos. It has about 1 thousand videos with HQ face swapping algorithms used to synthesize part of them. There are good quality synthesized videos with almost none artifacts of original face, small moving parts and other indicators. So it can be really hard challenge for the model.

TABLE II. AUC % performance on Celeb-DF dataset of our method

Model	Celeb-DF AUC
ResNet-50 + Gaussian blur	53.8
DenseNet169 + Gaussian blur	55.4
DenseNet169 + exponential blur	57.6
DenseNet169 + Rayleigh blur	**60.1**

TABLE III. AUC % performance on Celeb-DF dataset other methods

Model	Celeb-DF AUC
Two streams	55.7
HeadPose	54.8
FWA (ResNet50)	53.8
FWA (DenseNet169)	**60.1**

When we talk about video classification – we should get the point of very picky judge. Hence, when we decide, whether video modified or not – we split it frame by frame and analyze each one. If 1% of frames are classified as modified (face swapped) we should mark the whole video as Deepfake. Same can be said about photos or frames with multiple faces. We should mark the whole photo synthesized if any of the faces marked as fake.

VII. Comparing with state-of-the-art

We compare the AUC performance of our model. Table III shows the performance of 3 best methods of detection that were previously used on this dataset compared to our solution. As can be seen we outperform other methods with our model. However, we also could not get ideal results. High quality deepfake algorithms used to produce this data that is the reason of inconsistency of face warping artifacts indicator.

VIII. Conclusion

In this work, we describe a summary of indicators that can be used to decide, whether video or photo was changed with usage of face swapping AI-based algorithms (Deepfake). Our choice of building model is face warping artifacts detection that is one of the best indicators of fake video/photo right now. That conclusion bases on fact that great part of present deepfake algorithms can synthesize only low quality resolution faces. And then they should affine them to make picture whole and smooth. Such transformation leave distinctive artifacts, that can be detected. We evaluated our model on one of the newest and best datasets present right now and got pretty good results, that demonstrates effectiveness of method in practice.

As the offensive face swapping technologies keeps evolving, we should continue researches on effective methods of defense. Our research can be continued with exploration of new network structures for more efficient detection of Deepfake videos, or exploration of new indicators which can appear in new versions of Deepfake algorithms.

Acknowledgment

The authors are sincerely grateful to the head of the Department of Computer Systems and Technologies of the National Research Nuclear University "MEPhI" Professor M.A. Ivanov for help and support during the research. Research of statistical safety of stochastic transformation blocks was held within the framework of the Program of improving the competitiveness of the National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), as well as in the framework of the priority areas grant program of the Russian Science Foundation “Fundamental and exploratory studies by individual research groups”.

References

[1] faceswap github. https://github.com/deepfakes/faceswap (Accessed Nov 4, 2019)

[2] faceswap-gan github. https://github.com/shaoanlu/faceswap-GAN (Accessed Nov 4, 2019)

[3] Deepfacelab github. https://github.com/iperov/DeepFaceLab (Accessed Nov 4, 2019)

[4] Dfaker github. https://github.com/dfaker/df (Accessed Nov 4, 2019)

[5] DeepFake-tf github. https://github.com/StromWine/DeepFake-tf (Accessed Nov 4, 2019)

[6] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking. In IEEE International Workshop on Information Forensics and Security (WIFS), 2018.

[7] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. In IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[8] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi and Siwey Lyu. Celeb-DF: A New Dataset for Deepfake Forensics. arXiv preprint arXiv:1909.12962, 2019.

[9] Gao Huang, Zhuang Liu, Laurens van der Maaten and Kilian Q. Weinberger. Densely connected convolutional networks. In IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017.

[10] Maksutov A.A, Simonenko A.V., Shmakov I.S. Classifiers based on Bayesian neural networks. Proceedings of the 2017 IEEE Russia Section Young Researchers in Electrical and Electronic Engineering Conference, ElConRus 2017, 2017г.

[11] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.

[12] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Two-stream neural networks for tampered face detection. In IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1831–1839, 2017.

[13] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In IEEE Winter Applications of Computer Vision Workshops (WACVW), 2019.

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

[15] Davis E. King. Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research, 10:1755–1758, 2009.

[16] Matthias Kirchner and Rainer Bohme. Hiding traces of resampling in digital images. IEEE Transactions on Information Forensics and Security, 2008

[17] Alin C Popescu and Hany Farid. Exposing digital forgeries by detecting traces of resampling. IEEE Transactions on signal processing, 2005.

[18] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[19] 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.

[20] Agarwal S. and Varshney L.R.. Limits of deepfake detection: A robust estimation viewpoint. arXiv preprint arXiv:1905.03493, 2019.