Deepfakes Creation and Detection Using Deep Learning

Hady A. Khalil, Shady A. Maged
Department of Mechatronics Engineering, Ain Shams University, Cairo, Egypt
hadyayman1996@gmail.com, shady.maged@eng.asu.edu.eg

2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC) | 978-1-6654-1243-8/20/$31.00 ©2021 IEEE | DOI: 10.1109/MIUCC52538.2021.9447642

***Abstract*—Deep learning has been used in a wide range of applications like computer vision, natural language processing and image detection. The advancement in deep learning algorithms in image detection and manipulation has led to the creation of deepfakes, deepfakes use deep learning algorithms to create fake images that are at times very hard to distinguish from real images. With the rising concern around personal privacy and security, Many methods to detect deepfake images have emerged, in this paper the use of deep learning for creating as well as detecting deepfakes is explored, this paper also propose the use of deep learning image enhancement method to improve the quality of deepfakes created.**

***Index Terms*—deepfake, deep learning, Artificial intelligence , machine learning, tensorflow**

I. Introduction

Machine vision is advancing day after day in different fields from basic image detection software to automotive and robotics [1] [2] [3], one of the applications that stem from machine vision is Deepfake.

Deepfake is a technique that uses deep learning algorithms to create fake images usually by swapping a person's face from a source image into another person's face in a target image, with a resulting fake image that is sometimes hard to detect.

The underlying mechanism for deepfake creation is using deep learning encoders and decoders, which have been used extensively in the machine vision domain [4] [5]. the encoders work by extracting all the features in an image and then decoders are used to generate the fake image. deepfake methods need a large number of images and videos to train the deep learning models, this used to be a hard task but in the time that we live in you could easily find a large dataset of images on social media, this wide availability of data has led to the development of more complicated deepfake techniques, Many of deepfake algorithms are designed using Tensorflow [6]. TensorFlow is an open source software library for numerical computation using data-flow graphs. It was originally developed by Google to be used internally in it research and development of machine learning and deep neural networks, but the system is general enough to be applicable in a wide variety of other domains as well and it became very popular for machine learning applications after it was made available publicly and free to use.

TensorFlow provides a way to design neural networks quickly with adequate performance and APIs can be used with the python programming language, we can easily change the CNN architecture and test various designs without having to modify many lines of code.

Deepfakes pose a large threat to a future where fake news is everywhere, you could watch a video of an important public figure or president giving a speech and not be sure whether what you are seeing is real or fake [7], especially because the process of creating these fake images and videos is much simpler today as you only need as little as an image or a video of the target individual to generate the fake content.

Major tech companies are actively researching methods to detect deepfakes to fight the increasing number of deepfakes on the internet. Recently, Facebook, Amazon, Microsoft, and the Partnership on AI's Media Integrity Steering Committee have come together and launched the Deepfake Detection Challenge to encourage more research and development in detecting and preventing deepfakes [8]. Also, Google has released a free dataset for the public as a contribution to the deepfake detection challenge [9]. this interest in deepfake from large names like Google and Microsoft shows how significant the issue of deepfake.

In this paper one of methods of detecting deepfake images using Mesonet CNN is explored. The paper is structured as follows: section II deepfake creation, section III Deepfake Detection, section IV experimental Results, and finally concluding remarks are presented in section V.

II. Deepfake Creation

Deepfakes are created using deep learning methods in which they aim to replace the face of a targeted person by the face of someone else in an image or video. This technique was improved by developers and online communities to notably create user-friendly applications that are readily available online like FakeApp and FaceSwap [10] [11].

Deepfake relies on an autoencoder-decoder pipeline, encoders are widely used in image compression they rely on deep neural networks and by introducing a bottleneck in the network this forces a compressed representation of the original input [12]. as more advanced encoders are introduced, high-quality image compression is possible which can facilitate deepfake task as less computational power is required [13] [14]. deepfake creation work by training two autoencoders. One autoencoder learns the features of the source image and

the second encoder learns the feature the target image and two encoders share their parameters, then to generate the deepfake image, the target image is reconstructed using the source image's decoder, this will generate an image of the target with features of the source image Fig. 1. This method is the most common way of deepfake image creations and is used by DeepFaceLab [15] [16] and many others. The performance of this method is directly related to the size of the dataset available to train the deep neural network, luckily a huge number of datasets is readily available to the public on the internet in the form of social media, an incredible amount of images and videos is available publicly on instragram and youtube, especially for public figures and celebrities and that is exactly why public figures where the first target of deepfakes and are the ones most affected by it [17]. Also, the research community provides many datasets to use in deepfake.

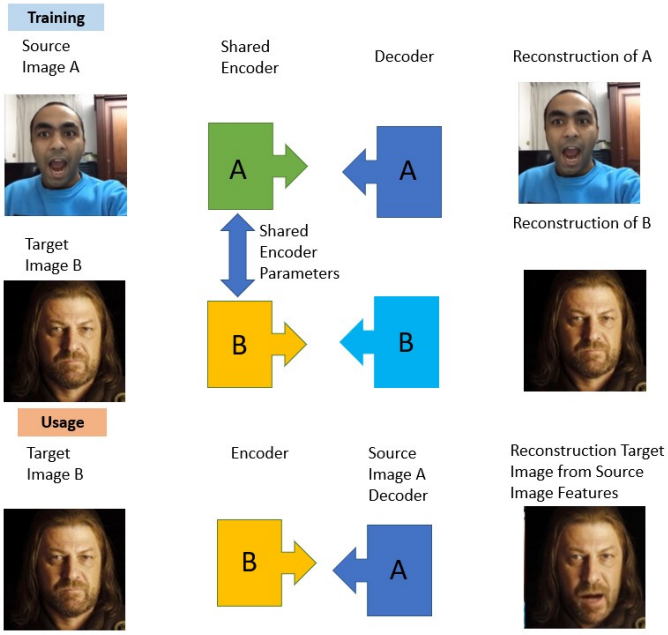

Fig. 1: Deepfake Creation Layout.

These principles can be used to generate deepfake images and/or videos, obviously, images are faster to generate as they require less processing and have a small size compared to videos. With the advancement in deepfake tools and the rise of fake news, the danger of deepfake is clear, hard to detect fake images Fig. 2. are becoming easier to make with user friendly deepfake tools and fake news that could utilize such images are widespread all over the internet.

That's why the need for tools to detect deepfakes is becoming very important day after day.

III. Deepfake Detection

Deepfake is commonly used to create swapped face images that are very challenging to detect [18]. Many proposed methods in the literature are available for both creating and detecting deepfakes [19].

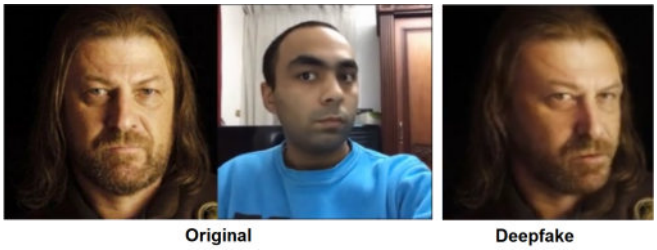

Fig. 2: Deepfake generated image.

D. Afchar et al. [20] proposed using a deep neural network with a small number of layers called MesoNet. the CNN demonstrated a very successful detection rate with more than 98 % for Deepfake.

A. MesoNet CNN

Mesonet is a neural network designed specifically to detect deepfakes. Deepfakes videos are usually found all around social media, the nature of videos on social media platform like Instagram is that they are low quality compressed video so microscopic analysis based on image noise is not possible and MesoNet takes that into consideration, also detecting deepfake on at a higher semantic level is hard as even humans sometimes struggle with detecting deepfakes [21], therefore MesoNet relies on an intermediate approach using a deep neural network with small amount of layers.

This network begins with a pattern of four layers of successive convolutions and pooling Fig. 3, and is followed by a dense network with one hidden layer. convolution and pooling are used to extract features of an image, it common pattern to use a convolution layer followed by a pooling layer as the convolution layer extract the features and the pooling layer creates a downsampled version of the feature map. To

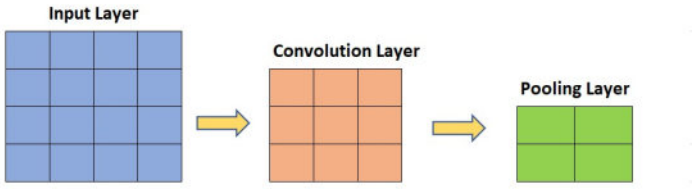

Fig. 3: Convolution and Pooling Layers Pattern used in CNNs to extract feature maps

improve generalization, the convolutional layers use ReLU activation functions that introduce non-linearities and Batch Normalization [22] to regularize the output, and the fully-connected layers use Dropout to regularize and improve their robustness.

IV. Experimental Results

A. Deepfake Creation

Many Deepfake creation methods exist in the literature, most commonly are face swapping and image animation

techniques. The technique that was used is called First Order Motion Model for Image Animation [23]. The model takes 2 inputs a source video and target image and it then animates the target image based on the source video Fig. 4.

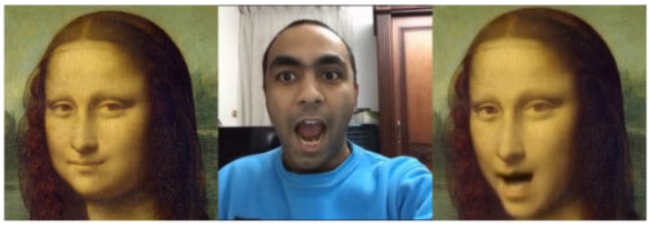

Fig. 4: deepfake video generated from source image

B. Deep Learning Image Enhancement

The use of image enhancing methods to improve the quality of created deepfakes was tested. First, the single-image super resolution method ESRGAN [24] was tested and while the resulting image was a higher resolution version of the input, there wasn't a drastic increase in the quality of the deepfake image that would justify this extra step. Then a blind face restoration method DFDNet [25] was used which had much better results, the produced image from DFDNet had a very significant increase in quality and resulted in a more realistic looking deepfakes Fig. 5. These methods were all tested using BasicSR [26], BasicSR (Basic Super Restoration) is an open source image and video restoration toolbox such as super-resolution, denoise, deblurring, JPEG artifacts removal, etc. It provides a quick and easy way to test image enhancement algorithms like ESRGAN, DFDNet, StyleGAN2, and many more.

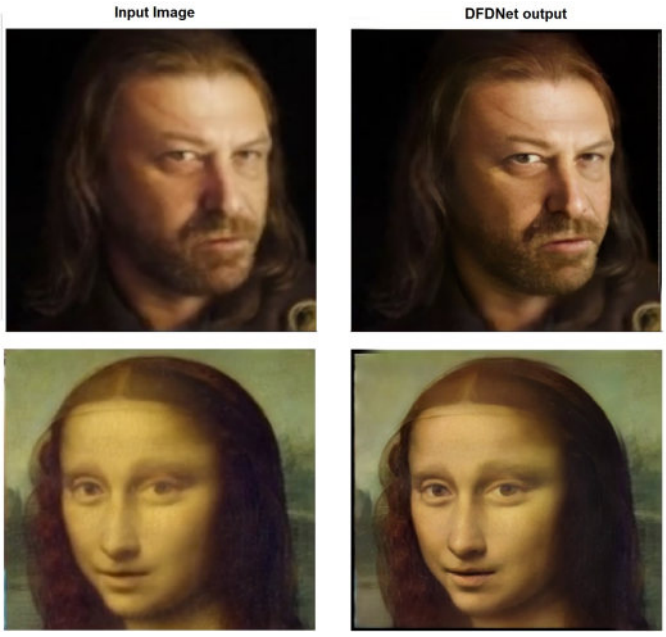

Fig. 5: Increasing quality of Deepfake with DFDNet

C. Deepfake Detection

To successfully build a robust neural network capable of detecting complex deepfakes a large dataset of training images is needed. due to social media these kind of images can be easily obtained also companies like google provide large datasets to help accelerate the research in protection against deepfakes . MesoNet used a dataset of more than 5000 images Fig. 6, the images are divided into real images and deepfake images.

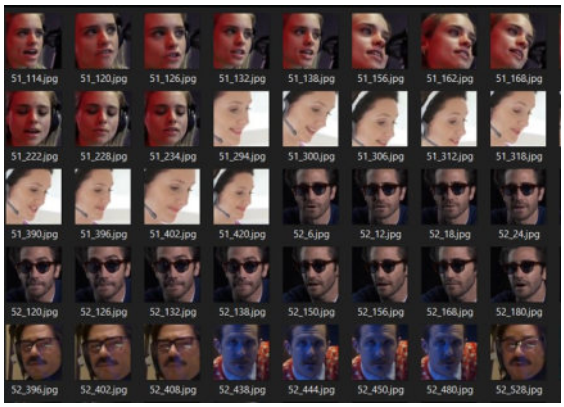

Fig. 6: dataset

After training the CNN with the dataset, the CNN is able to detect deepfake images with over 80 % confidence rate Fig. 7. When exploring current deepfake images available online, deepfake flaws are easily noticeable, deepfake images of a person's face looking straight ahead to the camera are harder to detect than images where the person's face is looking at an angle. current deepfake generation techniques aren't that great at dealing with faces at an angle.

Fig. 7: Correctly classified images with high confidence rate over 80%

but advanced deepfake generation techniques can sometimes generate images that are very hard to detect from real images, despite using a large dataset, some deepfake images are misclassified as real images Fig. 8.

V. Conclusion

In this paper deepfake creation and detection was explored as well as the integration of deep learning image enhancement

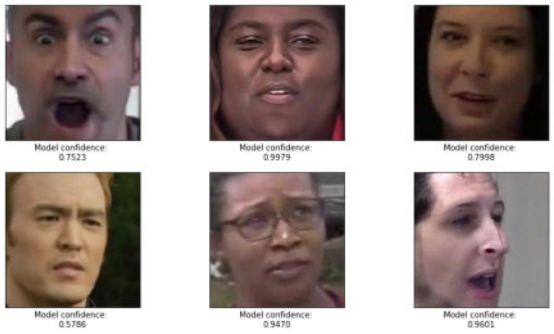

Fig. 8: Deepfake images Misclassified as real images

method to increase the quality of deepfakes created. From our experiment, the use of image enhancement methods like DFDNet gave higher-quality look deepfake images, which give them a more genuine look unlike the typically low quality associated with fake images on the internet. Deepfake detection in the case of face-swapping deepfakes, they are hard to detect when the person is facing straight towards to camera, when the person looks to his side, imperfections in the deepfake generated image can be seen. We believe that to further improve the performance of deepfake detectors, the focus should be on using datasets of difficult conditions like this. Our future work will explore generating deepfakes with reduced imperfections and higher quality using image enhancement methods to try and make them harder to detect for deepfake detection methods.

REFERENCES

[1] Wiley, Victor and Lucas, Thomas. (2018). Computer Vision and Image Processing: A Paper Review. International Journal of Artificial Intelligence Research.

[2] Bharathi, S.Shankar and N.Radhakrishnan, and Prasad, Pinnamaneni. (2013). Machine Vision Solutions in Automotive Industry.

[3] M. H. Wagdy, H. A. Khalil and S. A. Maged, "Swarm Robotics Pattern Formation Algorithms," 2020 8th International Conference on Control, Mechatronics and Automation (ICCMA), 2020, pp. 12-17.

[4] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence,39(12), 2481-2495.

[5] Yang, W., Hui, C., Chen, Z., Xue, J. H., and Liao, Q (2019). FV-GAN: Finger vein representation using generative adversarial networks. IEEE Transactions on Information Forensics and Security, 14(9), 2512-2524.

[6] TensorFlow. Accessed on: December 19, 2020. Available at https://www.tensorflow.org/

[7] AKaliyar, R. K., Goswami, A., and Narang, P. (2020). Deepfake: improving fake news detection using tensor decomposition based deep neural network. Journal of Supercomputing.

[8] Deepfake Detection Challenge Results. Accessed on: January 15, 2021. Available at https://ai.facebook.com/blog/deepfake-detection-challenge-results-an-open-initiative-to-advance-ai/

[9] Dolhansky, Brian and Howes, Russ and Pflaum, Ben and Baram, Nicole and Ferrer, Cristian.(2019). The Deepfake Detection Challenge(DFDC)Preview Dataset.

[10] Faceswap: Deepfakes software for all. Accessed on: February 2, 2021. Available at https://github.com/deepfakes/faceswap

[11] FakeApp 2.2.0. Accessed on: February 2, 2021. Available at https://www.malavida.com/en/soft/fakeapp/

[12] Ballé, Johannes and Laparra, Valero and Simoncelli, Eero.(2016). End-to-end Optimized Image Compression.

[13] Cheng, Zhengxue and Sun, Heming and Takeuchi, Masaru and Katto, Jiro. (2019). Energy Compaction-Based Image Compression Using Convolutional AutoEncoder. IEEE Transactions on Multimedia. PP. 1-1.

[14] A. Punnappurath and M. S. Brown, "Learning Raw Image Reconstruction-Aware Deep Image Compressors," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 4, pp. 1013-1019, 1 April 2020.

[15] Petrov, Ivan, et al. (2020). DeepFaceLab: A simple, flexible and extensible face swapping framework. arXiv preprint arXiv:2005.05535, 2020.

[16] Tewari, Ayush, et al. (2018). High-Fidelity Monocular Face Reconstruction Based on an Unsupervised Model-Based Face Autoencoder. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[17] Deep Fakes, Fake News, and What Comes Next. Accessed on: January 20, 2021. Available at https://jsis.washington.edu/news/deep-fakes-fake-news-and-what-comes-next/

[18] Korshunova, I., Shi, W., Dambre, J., and Theis, L. (2017). Fast faceswap using convolutional neural networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 3677-3685).

[19] Mirsky, Yisroel and W. Lee. "The Creation and Detection of Deepfakes." ACM Computing Surveys (CSUR). vol. 54 no. 1 , April 2021.

[20] D. Afchar, V. Nozick, J. Yamagishi and I. Echizen, "MesoNet: a Compact Facial Video Forgery Detection Network," 2018 IEEE International Workshop on Information Forensics and Security (WIFS), 2018, pp. 1-7.

[21] V. Schetinger, M. M. Oliveira, R. da Silva, and T. J. Carvalho. Humans are easily fooled by digital images. arXiv preprint arXiv:1509.05301, 2015.

[22] Ioffe, Sergey and Szegedy, Christian. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv preprint arXiv:1502.03167, 2015.

[23] Siarohin, Aliaksandr, Stéphane Lathuilière, S. Tulyakov, E. Ricci and N. Sebe. "First Order Motion Model for Image Animation." NeurIPS (2019).

[24] Wang X. et al. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In: Leal-Taixé L., Roth S. (eds) Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science, vol 11133. Springer, Cham.

[25] Xiaoming Li. et al. Blind Face Restoration via Deep Multi-scale Component Dictionaries. arXiv preprint arXiv:2008.00418, 2020.

[26] Xintao Wang and Ke Yu and Kelvin C.K. Chan and Chao Dong and Chen Change Loy, "BasicSR," 2020, Accessed on: January 15, 2021. Available at https://github.com/xinntao/BasicSR