Image Animations on Driving Videos with DeepFakes and Detecting DeepFakes Generated Animations

Yushaa Shafqat Malik
Department of Computer Science
Forman Christian College University,
Lahore, Pakistan
213478776@formanite.fccollege.edu.pk

Nosheen Sabahat
Department of Computer Science
Forman Christian College University,
Lahore, Pakistan
nosheensabahat@fccollege.edu.pk

Muhammad Osama Moazzam
Computer Science Department
Forman Christian College University
Lahore, Pakistan
213503339@formanite.fccollege.edu.pk

2020 IEEE 23rd International Multitopic Conference (INMIC) | 978-1-7281-9893-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/INMIC50486.2020.9318064

***Abstract*—The concept of image animation is to create a video or animation such that an object from an image is animated as per the motion of driving video. We plan to analyze with minor modifications of an existing framework which does this without any information beforehand about the object which is to be animated. To achieve that, we train our dataset on a set of images and videos for the objects of same category, for example (face, body, street views) etc. Some recent applications of neural networks (CNN) have proved to form realistic human heads. Realistic talking heads can be created by training the dataset of large number of images and videos. A source image of a person can be animated on target poses of a person (driving video), by keeping the appearance and body of the person. However, on the parallel side, there are advancements in the development of systems which are capable of detecting DeepFakes generated videos and animations as it is a crucial security concern. We did experiments on Image Animation to achieve talking heads, Image generations with conditional generative adversarial networks for DeepFakes Generations and the results were realistic. Moreover, we implemented a DeepFake Detector XceptionNet with minor modifications which achieved 95% accuracy on detecting DeepFakes. At last, we implemented a newly introduced technique in which the DeepFake generation is perturbed through which it can easily fool the deepfake detector. XceptionNet was able to achieve less than 30% accuracy on detecting DeepFakes generations when they were perturbed.**

Keywords*—DeepFakes, Image Animation, DeepFakes Generations, Detection of DeepFakes, GANS, Adversarial Attacks, Fooling DeepFake Detectors

I. Introduction

The concept of DeepFakes has been very popular and has applications in various fields. Image animation is also composed under DeepFakes. It is the concept of synthesizing a video by two major parts. First part is the driving video and the second part is the source image from which the object or the face/body is to be extracted. For instance, a person is moving his head and lips in a particular pattern, we can attach the image of another person (our source image) on that driving video. It is expected most likely very soon, in the film industry, there will be Artificial Intelligence Generated human like characters which wouldn't exist in real but will look exactly like a normal human on the screen and will play roles as movie characters. All these advancements are based on the advancements in the field of Artificial Intelligence, Deep Learning combined with Computer Vision.

Generative Adversarial Networks are capable of performing image animation and video retargeting [1,2,3]. Transferring of facial expressions is performed by Variational Auto-Encoders [3]. However, Aliaksandr et. al [4] came up with a better approach for this "Monkey-NET". Monkey-NET animates the source image according to the corresponding trajectories estimated in the driving video. Such advancements in the field of DeepFakes can be dangerous at times too [4].

DeepFakes, is a threat to Face Recognition [4]. The deception could be dangerous, as this kind of technology could be misused if provided in wrong hands. In this era of technology, it has become quite easier to swap face of a person with his/her single image into any driving video, which is extremely dangerous. With such menacing situation, it demands its solution too. There has been a research on the development of several software which are capable enough to detect Deep Fake generated videos/animations/images.

An audio-visual based approach has also been proposed which detects the patterns of lip sync with the visuals and the audio. However, this didn't seem to be very accurate. The most commonly used technique for detecting DeepFakes generations is Adversarial Network Attacks using Generative Adversarial Networks [1], which has been quite successful.

The paper is structured as follows. Section II shows the related work in this field. Section III shows our proposed methodology followed by experiments conducted in Section IV. Analysis and results have been presented in Section V. While Section VI concludes the research.

II. Related Work

The practical implementations of DeepFakes is in various fields. A very positive example of implementations of DeepFakes is a video, a campaign for a social cause "Malaria must die"[5]. Former football star, David Beckham is casted in that video ad where he speaks in 9 different languages, even in a female voice. All because of DeepFakes, that even while speaking in a different language, lip sync of David Beckham is perfect. Audio and video syncs totally fine.

This example suggests that DeepFake technology can be used in the Film Industry for dubbing purposes, and even portraying characters which were not actually in front of the camera, or even they never existed. Human like characters can be generated with such technology. However, it has its darker side too. It gained maximum attention when former president of United States of America, Barrack Obama's DeepFake

978-1-7281-9893-4/20/$31.00 ©2020 IEEE

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.

generated video leaked in public. Animating faces with one source images on driving videos of talking heads is now a very common thing, but with advancements in this field, animation of complete body is now being done. To achieve this, Open Pose library is widely used.

Guha et. al [6] proposed a neural network which has the ability to translate pose changes to the image space. In simple words, it has the capacity to synthesize an image on a particular target pose, without affecting the body, background or anything in the image.

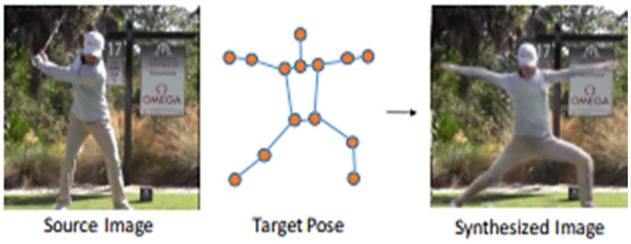


Fig. 1.   Example of Synthesizing source image onto a target pose [6]

Fig. 1 is the clear representation of Guha's work [6]. It can be observed that, initially there are two variables. Source image and a target pose. The modular structure of the network is capable of tackling maximum issues, as the pose transformation demands several challenges. This neural network is very much capable of video synthesis too. The collection of images having one source image and multiple target poses can form into a video too. Keras [7] is a new, popular deep learning library mostly used by deep learning enthusiasts. Initially released in 2015, it has been very popular over a few years. Moreover, they used keras and tensorflow-backend for the implementation of their research.

However, the work that we are interested in is the implementation of Image animations.

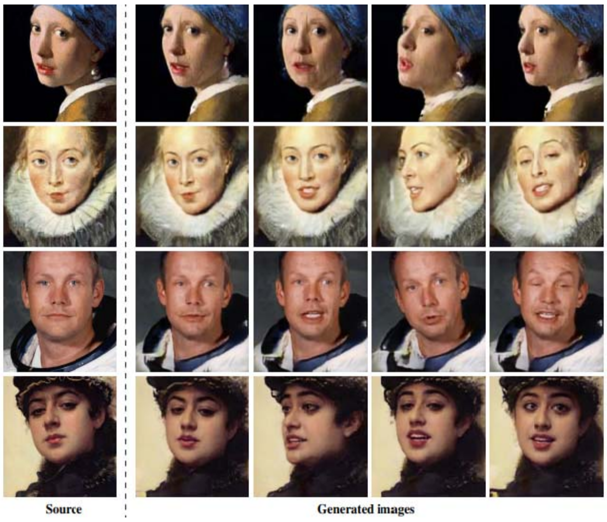


Fig. 2.   Example of Image Animation

Fig. 2 represents the implementation of image animation, how source images are animated on a driving video. The results are quite accurate as the framework proposed by Egor et. al [8]. Their work is the presentation of a framework of meta-learning Generative Adversarial Deep Networks which are highly capable to create Image animations for talking heads. The two datasets used in this paper are VoxCeleb (256p videos at 1fps) [9] and VoxCeleb2(224p videos at 25fps) [9]. These two have been very popular datasets lately for DeepFakes implementations.

On the contrary, a lot of work has been done on the detection of DeepFakes generations. Recently, a research has been done for the detection of DeepFakes detectors. Literature reveals that traditional adversarial attack using Generative Adversarial Networks are also implemented and the results are quite impressive [1][10].

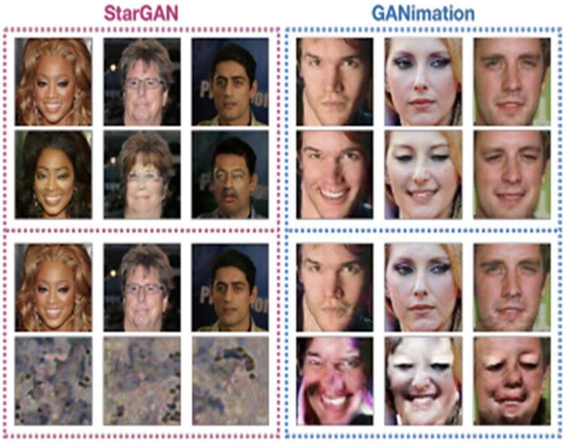


Fig. 3.   Example of Detecting DeepFakes

Fig. 3 is the representation of [10]'s detecting Deepfakes from the images. The adversarial attacks expose the deepfake generations and highlight the transformed area. These adversarial attacks are done on starGan and GANimation, two very popular techniques for DeepFakes generations.

These are a combination of two neural networks which sort of compete with each other [1]. The concept is much appreciated by the experts due to its capabilities. These are capable of generating new stuff. It is also considered as the most interesting idea in the field of AI in last ten years.

However, it is still a challenge as the Deepfake generation technology has been improving a lot, and it is even capable of fooling the detectors.

III. Methodology

This section contains a detailed description of our proposed methodology. We have tested our custom input on the Image Animation framework. We made several changes in the existing framework. While training, we set the betas to (0.75, 0.999) to optimizer generator, discriminator, and Detector. We also run the model on different number of epochs. While making animation by setting the variable "Relative" to false, the results weren't quite good. However, when we made animation by cropping the video and setting the variable "Relative" to true, the results proved to be better.

While detecting DeepFakes from an input video, we implemented XceptionNet [11] a popular deep network for detection of DeepFakes. We made 15 layers, with "relu" as activation function at each layer. This detector proved to generate quite accurate results on the DeepFake generated media.

We also implemented the existing model of Adversarial Attacks on DeepFake generations to fool the deepfake detectors.

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.

A. *Data Collection*

VoxCeleb [9] has been a very popular dataset. It is gloabally available and is open source. Basically, it has two versions, VoxCeleb1 and VoxCeleb2. The dataset is created after a lot of efforts, capturing interviews of celebrities and famous personalities from one of the most famous websites, YouTube. The dataset mostly comprises of famous personalities belonging from the United States of America. the VoxCeleb1 has over 100,000 samples and VoxCeleb2 having over a million samples, without any overlapping in between these two datasets. This dataset has been utilized in the field of Computer Vision for the implementations of projects like, Emotion recognition, Speaker Identification, Face generation and Talking Face Synthesis. Our main focus is on Face generation and Talking Face Synthesis for this research.

There is a dataset dedicated for Pix2Pix dataset, available on Kaggle [12] for the implementations of Generative Adversarial Networks. We have used FaceForensics++ [13] which is an open source publicly available dataset [16]. This dataset is completely dedicated for the detection of manipulated facial images. It contains 1000 video sequences which are generated by four different manipulation methods as follows:

1) *DeepFakes*
2) *Face2Face*
3) *FaceSwap.*
4) *NeuralTextures*

IV. Experimentation

In this section we briefly explain the experiments carried out during our research.

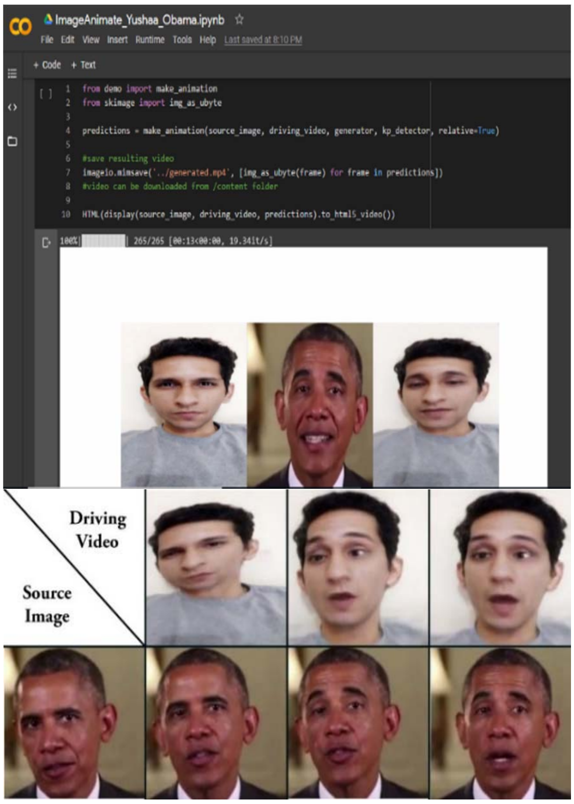
In the first step, we animated a source image on a driving video, and made talking heads. Then we implemented cGans by giving two inputs, target image and several source images and constructed a new predicted image, and the results were very realistic. During our research we also detected DeepFake generated images and videos, which proved to work efficiently in limited cases, however when we applied adversarial attacks on our DeepFake generated videos, we saw that the DeepFakes detector was fooled.

In the following sections, the experiments conducted are discussed.

A. *Image Animation*

In this research, DeepFakes Image animation is implemented. For coding purpose, due to low processing power available, Google Colaboratory (Service provided by google), is used, to utilize GPU and more processing power to run the model to write Notebook. The dataset and framework files are uploaded on the Google Drive as it is directly accessible from Google Colaboratory.

We define source image and driving video, and simply display them. We further load demo.py and import load checkpoints and set it as vox-256.yml. Finally, we import make animation function form our demo.py file and animate the source image on the driving video as it can be seen in the Fig. 4 below.


ImageAnimate_Yushaa_Obama.ipynb
File Edit View Insert Runtime Tools Help Last saved at 8:10 PM
+ Code + Text
Driving Video
Source Image

```
from demo import make_animation
from skimage import img_as_ubyte

predictions = make_animation(source_image, driving_video, generator, kp_detector, relative=True)

#save resulting video
imageio.mimsave('../generated.mp4', [img_as_ubyte(frame) for frame in predictions])
#video can be downloaded from /content folder

HTML(display(source_image, driving_video, predictions).to_html5_video())
```

```
100%|          | 265/265 [00:13<00:00, 19.34it/s]
```

Fig. 4. Example of our output without crop

Since there can be variations in the driving video and we have on still source image, the results can misbehave sometimes. To tackle this, we use the source images appropriately and crop the driving video on appropriate proportions. Fig. 5 shows how to crop the video properly, and the results are quite better.

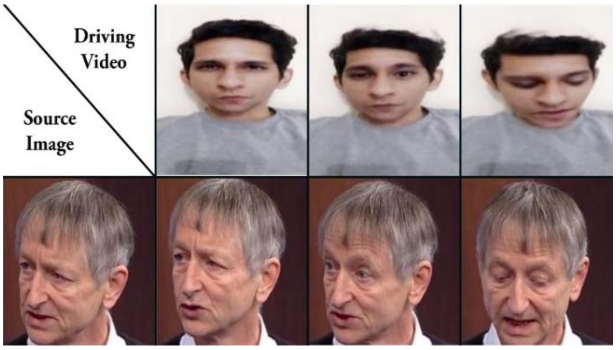


Fig. 5. Example of our output with crop

It is observed that the concept of Image Animation [3] shows quite amazing results. Irrespective of the source image, the source image is animated into talking head, very close to the driving video. Even if the user is having issues with the driving video, this problem is also resolved by cropping the video in appropriate proportions for better results. The framework was tested on three different datasets and it proved itself to work appropriately in all conditions.

B. *Image To Image generation with cGans.*

Phillip et. al [14] showed the implementations of conditional generative adversarial networks for image to image generation. So, in this experiment, we tried to implement his proposed conditional generative adversarial

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.

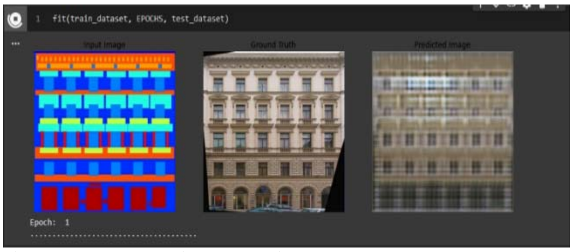
networks by providing an input image, target image and achieved the predicted image, and the results are quite amazing. We used pix2pix dataset for these experiments.

Having multiple source and target images, a new generated image is predicted which tends to be very realistic. We use TensorFlow, very popular deep learning library powered by Google for the implementations.

Fig. 6 shows the initial stage, as one input and one source image, and the predicted image are displayed. We plan to run our model on 150 epochs.

The snapshot in in Fig. 7 is taken in between, after completing 41 epochs and we can clearly see how the predicted image and the ground truth are looking very similar.

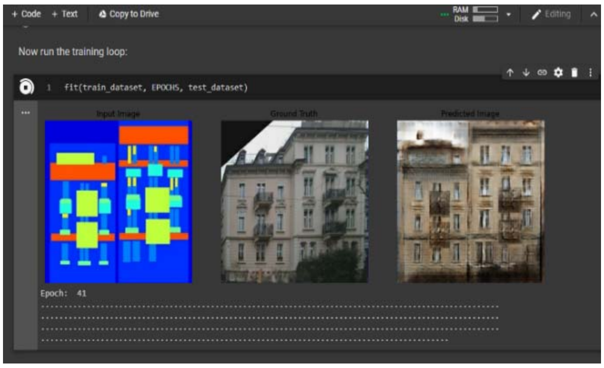
Fig. 8 shows the results achieved after 150 epochs. Each epoch took almost 30-35 seconds while executing the code on Google Colaboratory with GPU accelerated.



```
fit(train_dataset, EPOCHS, test_dataset)
```

```
Epoch:  1
```

Fig. 6. Image generation at Epoch 1



Now run the training loop:

```
fit(train_dataset, EPOCHS, test_dataset)
```

```
Epoch:  41
```

Fig. 7. Image generation at Epoch 43

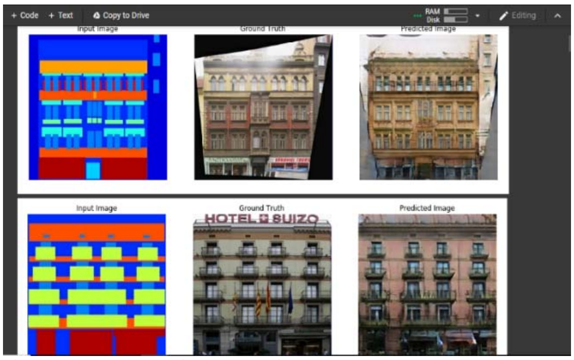


Fig. 8. Image generation at Epoch 150

C. Detecting DeepFakes

Li et. al [15] worked on CNN based model VGG-16 and ResNet to introduce a method which can easily detect DeepFake generated videos. Taking the advantage of the fact the DeepFake generations generate a limited resolution images only, this model is very much capable of detecting DeepFake generated images. While the generations of DeepFake image, go under some transformations which causes artifacts on the face. Therefore, this method takes advantage of the transformation and then the CNN based detectors detects the DeepFake generated images. An intelligent technique for training dataset is to not only include all fake images, instead mix the dataset with original images and DeepFake generated images. However, out of CNN-VGG-16, ResNet-101 and ResNet-152, ResNet-101 proved to perform the best for the detection of DeepFakes, as it scored 99% AUC.

D. Fooling DeepFake Detectors

A lot of research and practical implementations on the detection of DeepFake generations can be seen in literature. Many deep learning based software have been introduced, and most of them proved to be very accurate when implemented on CNN based DeepFake generated images.

However, we implemented the concept discussed by Paarth et.al [16] to fool DeepFake detectors which rely on CNN Architecture as mentioned in [15]. We use the dataset FaceForensics++[13] and tested our implementation on the famous algorithm XceptionNet[11] which is used to detect DeepFake images and videos, particularly for face-swapping detections.

XceptionNet[11] has proved itself to be very precise on detecting face swapping conventional techniques like FaceSwap(FS), Face2Face, DeepFakes and NeuralTextures having 90% above accuracy [11].

Considering CNN based DeepFake detectors as our target victim, it can be observed that these detectors scan each frame and see if it is real or fake. It can be difficult to fool such detectors as the conventional DeepFake models lack the originality in lip movement, eye blinking and even head movement.

The idea is to craft each frame of the video adversarial to bypass the DeepFake Detectors. We tested a video from FaceForensics++ [13] on XceptionNET and it was able to detect it as fake as shown in Fig. 9.

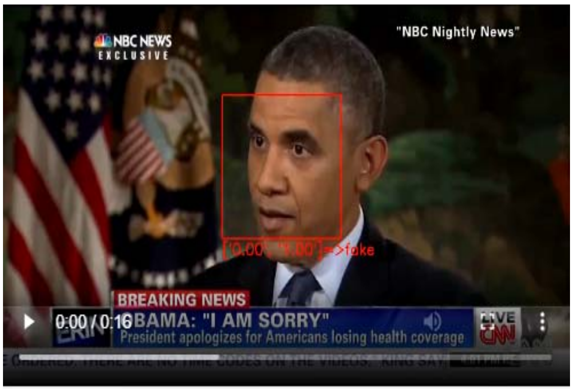


Fig. 9. DeepFake detected by DeepFake Detector

But after implementing adversarial effect on each frame of the same video using Generative Adversarial Networks, we can see how easily the XceptionNet DeepFake detector is fooled. It is evident that DeepFake detectors which rely on CNN-Architecture can be easily fooled by implementing Adversarial attacks on the test video. Fig. 10 shows Deepfake fooled by DeepFake detector.

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.

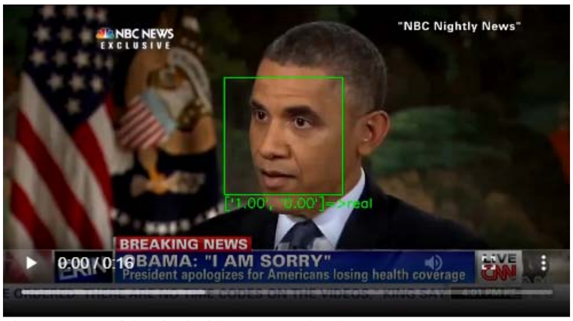


Fig. 10. Deepfake fooled by DeepFake detector

V. Analysis and Results

In this section we discuss the experiments performed during the research and their analysis. We changed the values of betas to (0.75, 0.999) while training the model of the framekwork [3], and also changed the variable "relative" to true while cropping the video and the model proved to work just fine on custom input to animate talking heads.

We observed various results by changing the number of epochs while implementing [14] Image to image translation using [12] pix-to-pix dataset. Results are mentioned in this paper at number of epochs 1, 43, and 150. Although we believe that increasing the number of epochs to 200 or 250 with more target images might generate a better predicted image.

We couldn't make any changes in the existing technique to fool DeepFakes [16] but discovered that XceptionNet [11] achieved less than 30% accuracy while detecting the DeepFake video. Video reconstruction comparison on different datasets gave us the following results as shown in Table I.

TABLE I. Video Reconstruction Comaprison On Different DataSets

	TaiChi	VoxCeleb	Nemo
L1	0.066	0.041	0.016
AKD	6.872	1.297	1.19
AED	0.176	0.140	0.046

While implementing XceptionNet [14], we changed the number of layers to 10, 12 and 15, and Relu as activation function at each layer, and we can see that the model with 15 layers produced best results achieving 95% accuracy to detect DeepFakes generations. Table II shows the comparison of Xceptionnet and Inception v3.

TABLE II. Comparison Of XceptionNet and Inception v3

Technqiue	Accuracy
Inception V3	94%
XceptionNet	95%

Table III shows the accuracy of DeepFakes detector. It can be observed that unperturbed DeepFake Generations gave more than 95% accuracy.

TABLE III. Accuracy of DeepFakes Detector

DeepFake Generations	Accuracy
Unperturbed	Greater than 95%
Perturbed	Less than 30% (Negligible)

Table IV shows the detection of DeepFakes. According to these results, it can be observed that CNN based DeepFake generations were not detected by DeepFake detectors which rely on Audio-Visual sync. However, CNN based detectors [15] were able to detect DeepFakes generations. But when DeepFakes generations [16] were perturbed, the detector couldn't detect them.

TABLE IV. Detection of DeepFakes

Architecture	Detection Method	Status
CNN	Audio-Visual	Not Detected
CNN	CNN	Detected
Adversarial Attack	CNN	Not Detected

VI. Conclusions And Future Work

DeepFakes technique is used in several fields. Despite of having its positive implementations in the field of Film Industry, maximizing the production with minimum input, it is still a very dangerous technique and a threat to the world. Manipulative content has been, and still being created with this technology. It is now very easier to defame any famous person by face swapping, or even whole body can be swapped.

This research intends to show the implementation of DeepFakes technology to swap a face, and even complete body with exact same background onto another driving video or a sequence of different poses. By various experiments, it is shown that how possible it is to swap a face of a particular person an animate it as a talking head onto a totally different driving video of another person and make them look like its actually them.

Moreover, we have seen that research is conducted on detecting these fake manipulative videos and images too which proved to be highly accurate. However, they have their limitations as well. CNN-Based detectors worked quite fine, as they classified each and every frame as original or manipulated but even these detectors failed after the DeepFake videos has Adversarial attacks.

Therefore, after performing so may experiments, we conclude that there is a decent progress in the development of DeepFakes detectors and Generative Adversarial Attacks/Networks might be able to detect DeepFakes media the best.

In future, we intend to modify the existing Adversarial attacks on Deepfake generated media to fool the DeepFake detectors more efficiently, and then do a proper detailed, in depth research on how to detect a perturbed DeepFake video so that even perturbed DeepFake generated videos or images can be detected.

References

[1] A. Bansal, S. Ma, D. Ramanan and Y. Sheikh, "Recycle-GAN: Unsupervised Video Retargeting", Computer Vision – ECCV 2018, pp. 122-138, 2018. Available: https://arxiv.org/pdf/1808.05174.pdf. [Accessed 10 July 2020].

[2] A. Nagrani, J. Chung and A. Zisserman, "VoxCeleb: A Large-Scale Speaker Identification Dataset", Interspeech 2017, 2017. Available: 10.21437/interspeech.2017-950 [Accessed 10 September 2020].

[3] A. Rosebrock, "Building a simple Keras + deep learning REST API", The Keras Blog, 2018. .

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.

[4] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies and M. Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images", IEEE/CVF International Conference on Computer Vision (ICCV), vol. 3, pp. 1-11, 2019. Available: 10.1109/ICCV.2019.00009 [Accessed 9 July 2020].

[5] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci and N. Sebe, "First Order Motion Model for Image Animation", NeurIPS, 2020. Available: https://www.researchgate.net/publication/339642276_First_Order_Motion_Model_for_Image_Animation. [Accessed 10 July 2020].

[6] Balakrishnan, G., Zhao, A., Dalca, A.V., Durand, F., & Guttag, J. (2018). Synthesizing Images of Humans in Unseen Poses. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8340-8348.

[7] E. Zakharov, A. Shysheya, E. Burkov and V. Lempitsky, "Few-Shot Adversarial Learning of Realistic Neural Talking Head Models", arXiv, vol. 2, no. 190508233, 2019. [Accessed 10 September 2020].

[8] F. chollet, "Xception: Deep Learning with Depthwise Separable Convolutions", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1800-1807, 2017. Available: 10.1109/CVPR.2017.195. [Accessed 10 September 2020].

[9] I. J. Goodfellow et al., "Generative adversarial nets", arXiv, vol. 2, pp. 2672–2680, 2014. [Accessed 9 July 2020].

[10] M. Kalmykov, “Positive Applications for Deepfake Technology”. 12-Nov.-2019.

[11] N. Ruiz, S. Adel Bargal and S. Sclaroff, "Disrupting Deepfakes: Adversarial Attacks Against Conditional Image Translation Networks and Facial Manipulation Systems", arXiv, vol. 3, no. 200301279, 2020. [Accessed 7 July 2020].

[12] P. Isola, J. Zhu, T. Zhou and A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks", arXiv, pp. 5967-5976, 2017. Available: 10.1109/CVPR.2017.632 [Accessed 10 June 2020].

[13] P. Korshunov and S. Marcel, "DeepFakes: a New Threat to Face Recognition? Assessment and Detection", arXiv, no. 181208685, 2018. Available: https://arxiv.org/pdf/1812.08685.pdf. [Accessed 10 June 2020].

[14] P. Neekhara, S. Hussain, M. Jere, F. Koushanfar and J. McAuley, "Adversarial Deepfakes: Evaluating Vulnerability of Deepfake Detectors to Adversarial Examples", arXiv, vol. 200212749, 2020. Available: http://10.13140/RG.2.2.26227.48168. [Accessed 10 June 2020].

[15] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts", arXiv, vol. 200212749, 2018. [Accessed 11 July 2020].

[16] “pix2pix dataset | Kaggle”. [Online]. Available:https://www.kaggle.com/vikramtiwari/pix2pix-dataset. [Accessed: 10-Sep.-2020].

Authorized licensed use limited to: Ben-Gurion University of the Negev. Downloaded on June 03,2022 at 11:43:46 UTC from IEEE Xplore. Restrictions apply.