International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

Comparative Study of Models on Fake Image Detection Generated by GANs

Dr Shailender Kumar[1]
Department Of Computer Engineering
Delhi Technological University
NCT of Delhi, India

Nand Kumar[3]
Computer Engineering
Delhi Technological University
NCT of Delhi, India

Rakshit Singh[2]
Computer Engineering
Delhi Technological University
NCT of Delhi, India

Mukul Singh[4]
Computer Engineering
Delhi Technological University
NCT of Delhi, India

Abstract: The imminent challenge we face as a society is how to judge the authenticity of online content, be it machine learning generated pictures or videos, with the emergence of Generative Adversarial Network (GAN) and other deep learning based DeepFake techniques. We are confronted with an unparalleled risk of serious violations of basic human rights, as well as a profound, unavoidable shift in how people interact online. We've seen many cases of fraud and abuse of news headlines, medical (mis)information, and invasions of privacy. The aim of this proposed project is to detect DeepFake images using an online image database. The aim of this paper is to use convolutional neural networks to classify real versus fake images from a big online database. We wanted to see how well different convolutional neural networks (VGGFace, DenseNet, Inception, Resnet and Custom CNN Architecture) behaved. Future work will include using unsupervised clustering methods / auto-encoders to see whether real and fake images cluster separately, as well as using CNN visualisation methods to add clarity and interpretability to our models.

Introduction

There are many devices accessible in image altering and image control, which will change our real image. By naked eyes, we don't get which picture is unique and which picture is fake. In social media, all images aren't the right images. In particular, improvements to brilliant devices like cell phones assume an indispensable function in transferring and downloading pictures to those social networks. The social network is a platform where people mingle, offer, and spread information. Once in a while, the pictures give us wrong information. When the most control by photoshop or some other altering programming is a photograph alter has numerous procedures for controlling a picture utilizing a particular reason. Pictures work can be made by counterfeit for the different purposes it utilized that's why we need exact information.

DeepFake, which was an unheard concept until 2017 now is all over the social media. DeepFake creates manipulated images, recordings and sound by joining supervised Machine Learning (ML) techniques, for example, convolutionary neural networks (CNNs) and generative adversarial networks (GANs) with unaided ML strategies like autoencoders. This can prompt a huge number of moral predicaments with expansive social ramifications. This may incorporate the development of upsetting fake porn based on superstar data, the conveyance of fake news, and the duplicity of safety frameworks that depend on facial recognition frameworks. We used the"140000 Real and Fake Faces" kaggle data set [2] which comprises of 70,000 real faces (from Flickr) and 70,000 fake faces (StyleGAN-produced). For extra samples, we used the kaggle data set "Real and Fake Face Detection" [3]. To execute and tackle the current issue, we utilized four existing CNN systems (VGGFace, DenseNet, Resnet101 and Inception) alongside a custom CNN with the essential goal of recognizing genuine and fake data. Actually generative models learning based, for instance, variational autoencoders and generative antagonistic networks (GANs), has been comprehensively used to join the photograph practical halfway or entire substance of a photo or a video. Furthermore, late changes of the GANs, for instance, Progressive Growing of GANs (PGGAN)

and BigGAN, have been used to join a significantly photorealistic picture or video, which is hard to see as a fake by individuals in a confined time. At the point when everything is said in done, the generative applications perform picture understanding tasks, which can cause critical issues if a fake picture is improperly used through online media networks. Besides, the GANs could make a discourse video with the orchestrated facial substance of any popular legislator, making extreme issues the general public, political, and business exercises. Subsequently, a compelling fake face picture location strategy is earnestly required.

In our work, we will attempt to analyze various methods that are used in image to image translation. We will probably comprehend if, to which degree, and in which conditions, these assaults can be disclosed. To this end, we will think about a few arrangements, put together both with respect to cutting edge techniques taken from the picture criminological writing, and on universally useful profound convolutional neural networks (CNNs) that are appropriately prepared for this undertaking. This is both the most well-known and most testing circumstance since the pressure regularly performed upon picture transferring will in general hinder the presentation of phony indicators.

Background

DeepFake, made from a combination of words Deep learning and fake, is a machine learning method that can layer images or videos of a target person to that of another person to create a new “unreal” image/video of the target person doing or saying things the source person does. The construction of a DeepFake image or video in essence requires auto-encoders, which consists of an encoder and decoder. The process of creating a DeepFake first involves reducing the image to a decreased dimension space (compressed image), which retains critical image information by the encoder. The decoder is the next phase, and it recreates the image.

The fundamental DeepFake system now includes the use of generative adversarial networks (GAN), which are generative models that learn the distribution of data without supervision. GANs are an updated framework for estimating generative models using an adversarial approach in which two models are trained at the same time. DeepFakes are decoded using GANs, with the decoder consisting of G and D training in an antagonistic relationship. Because the generator creates new images from the latent representation of the original source, it must be constantly corrected as the discriminator attempts to determine whether or not the image will be formed. As a result, the decoder is very good at preserving significant image data in the latent space. Hence, a generator is produced that generates images that are extremely similar to real data, as well as a process where any defects are detected by the discriminator. When GANs were first introduced to DeepFake models, they had an architecture that included just one GAN. Since then, several advancements have been achieved with the intention of improving the quality of fake data. Among them is the use of Cycle-GANs to resolve the issue of several training models' need for paired training images during the training period [5]. While the realism of GAN images improves over time, the lack of control in DeepFake methods' performance, i.e. modifying explicit features such as stance, face form, and haircut in a picture of a face, remains a challenge. To address this problem, NVIDIA created the Style-Based Generator Architecture for GANs (StyleGAN) process [6]. The fake image is created by StyleGANs in stages, beginning with a low resolution and progressing to a high resolution (10241024). By changing the input of each layer separately, it manages features that are constituted in that layer, from features like pose and shape of face to fine details like hair colour), without affecting other layers.

The negative consequences of DeepFake have been widely felt, resulting in a slew of debates. The vast majority of DeepFake's identified targets are celebrities and politicians. One such example is a viral DeepFake video of Manoj Tiwari reaching out to people of Delhi during elections campaigning in three different languages. However, it was later confirmed that the video was made by his organisation itself with his consent. Fake celebrity

pornographic images, revenge porn, and malicious conspiracies have all been made using DeepFakes like FakeApp, OpenFaceSwap, and MyFakeApp.

The first attempts at designing DeepFake detection systems centered on the shortcomings in DeepFake generation methods, such as models that had not been trained on footage of people with their eyes closed. In their produced images, this resulted in abnormal blinking patterns [9]. However, this detection method was soon overcome by the next generation of DeepFake models which then included blinking in their training data.
The fact that the current DeepFake algorithm could only produce images of limited resolutions, which needed to be further warped to match the original faces in the source video, was also exploited by detection techniques. Such changes left distinct objects in the resultant DeepFake videos Li et al. demonstrated that CNNs can effectively capture these objects, which can then be used to differentiate between real and fake data. [10]
The use of head poses as a means of detecting inconsistencies in DeepFake photos has also been a priority [11]. Yang et al. developed a method based on the assumption that DeepFakes are produced by splicing a synthesized face region into the original image, adding errors that can be identified when 3D head poses are estimated from face images. The authors report an SVM classifier which was evaluated using a set of real face images and Deep Fakes after features were selected which cue specific features of head poses.

Models

1. **Custom CNN**
   Based on a tensorflow backend, our custom CNN uses 6 layer of convolutional layers, each of them which are paired with a batch normalization and maxpooling layer. Dropout was applied to each layer and the activation function used was Rectified Linear (ReLU). This decreases the chance of over fitting the data. Padding was also used in each layer with the addition of dense layer at the end of the network with sigmoid used as activation function. In addition to this, an alternating dilation rate of 2 and 4 in the layers so that there is spacing between the values in a kernel.

2. **VGG**
   Using the pretrained model of VGGFace network this model used five convolutional layers each paired with maxpooling layers. We used the Adam as the activation function. The first two blocks consist of two whereas others contain three convolutional layers each followed by a maxpooling layer. As VGGFace is pretrained we fine-tuned the network by adding a dense layer that helps in providing us the final features of the network. At last, a dense layer with sigmoid activation was used.

3. **DenseNet**
   In this network we used the pretrained network of DenseNet-121 network found in Keras Module. For fine tuning we added a dense layer as the last layer. Training was done with 100,000 images with 20,000 images for validation. Model used dense blocks of layers of batch normalization, 3X3 conv and Adam activation. The model also consists of 2X2 pooling layer and ending the network with a dense layer of sigmoid activation.

4. **InceptionV3**
   Used the pertained model of InceptionV3 network of InceptionV3 network found in Keras. Using the same architecture as in the DenseNet network mentioned above. Again fine tuning by adding a dense layer and followed by the maxpooling layer. Adam optimizer found in keras was used as the optimizer. This model was trained on 102,041 images again using 20,000 images used in validation. The network ends with a dense layer with sigmoid activation.

5. **Resnet101**
   Used the pertained model of InceptionV3 network of Resnet network found in Keras. Using the same architecture as in the DenseNet network mentioned above. This model was also trained on 102,041 images again using 20,000 images used in validation. The network ends with a dense layer with sigmoid activation.

6. **Data Augmented Models**
   In all the models we mentioned above (except VGG) we have paralleled data

augmented models also to study the effect of augmentation on the models used. For data augmentation we flipped all the images horizontally. In addition to this rescaling was used to ensure all numbers are in the RGB range so that the image quality is maintained as all images do not have the same pixels.
Also, rotation was introduced with 20 and both shear and zoom range was also introduced with both being 20% of the original image.

7. **PCA+SVM**
For all the models that are mentioned above we extracted the last layer before classification to see the representation of images as vectors. The vectors were relatively big so we used Principal Component Analysis to keep the data that was most viable to contribute. Therefore by running PCA we were able to keep 100 principal components. We then were able to use Support Vector Machine with polynomial Kernel to classify the components as real or fake.

8. **Densenet with augmented Data**
The main motive of this model is to see the effect of if the colour has an effect on the accuracy of the networks. In this model we used the same network as in original Densenet but changed the pixels so that nothing is in the range of RGB.

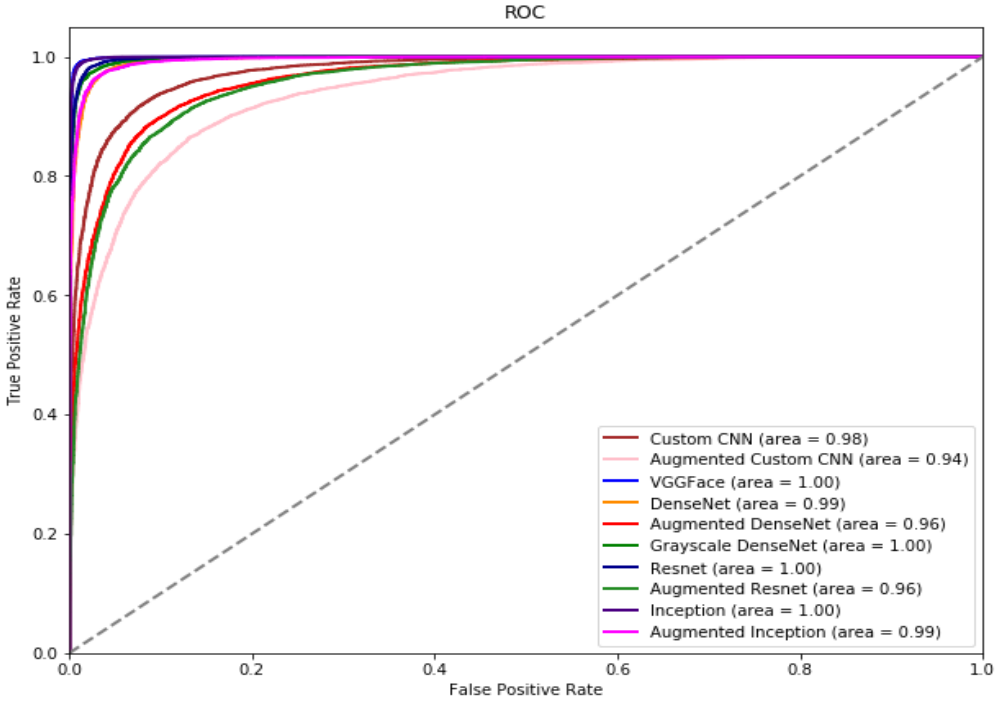
Results
Through this comparative study we can clearly see that the pre trained models are very efficient in classifying images that are generated by GANs.

Model	Neural Networks			SVM after PCA		
Metrics	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Custom Model	0.91	0.91	0.91	0.96	0.95	0.96
Custom with Aug Data	0.82	0.85	0.82	0.89	0.89	0.89
VGG	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
DenseNet	0.95	0.95	0.94	0.97	0.97	0.97
DenseNet with Aug Data	0.85	0.88	0.85	0.90	0.90	0.90
ResNet101	0.98	0.98	0.98	0.98	0.98	0.98
ResNet101 with Aug Data	0.89	0.89	0.89	0.89	0.89	0.89
InceptionV3	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
InceptionV3 with Aug Data	0.96	0.96	0.96	0.96	0.96	0.96
DenseNet with Grayscale	0.97	0.97	0.97	0.50	0.50	0.64

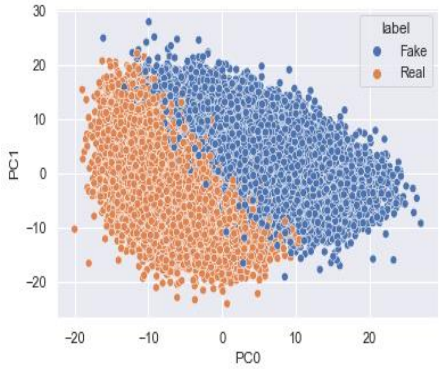
We can see clearly that the both VGGFace Architecture and InceptionV3 gave an accuracy of 99%. VGG however require very sophisticated processors and GPUs for its training on augmented data. Resnet also gives almost similar result though is more computationally extensive than InceptionV3. Densenet also gives decent result and is less computational than the models above. Introducing data augmentation, naturally there is decrease in the accuracies of the models. Although the model of InceptionV3 gives the most decent result on both the normal and augmented data.

With the last model we can see the colour have no influence on the accuracy of the model. Thus colour is not an attribute that can affect the performance of the models.

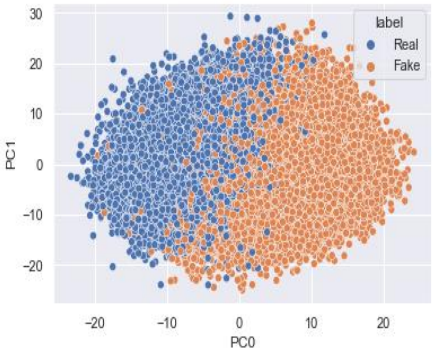
The custom model with diluted convolutional layers gave decent results although the accuracy was the least of all models. Also the introduction of augmented data comes with a sharp decrease of accuracy of 9%.ROC plot of all models is given below:-



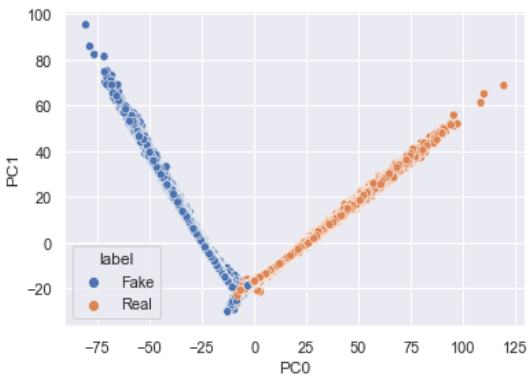
The scatter plot of all the models accept the model 8(Densenet with grayscale) does not form very distinctive clusters resulting in very poor performance as we can see in the table above. Other than this model the graphs give very distinctive clusters and we can see a normal increase in the accuracies of all models.



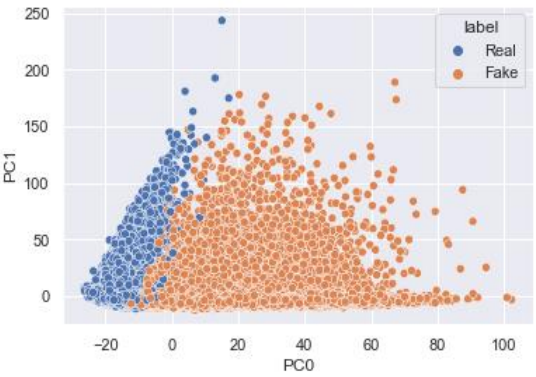
Custom CNN



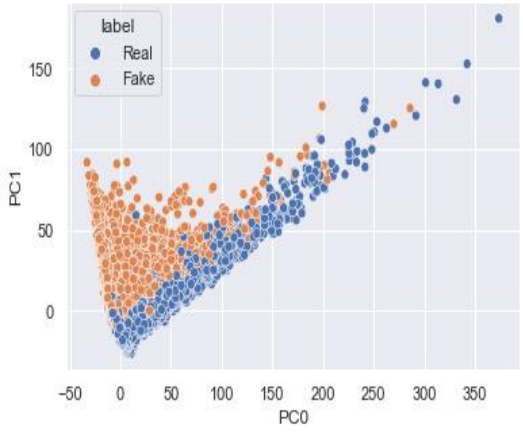
Custom with Aug Data



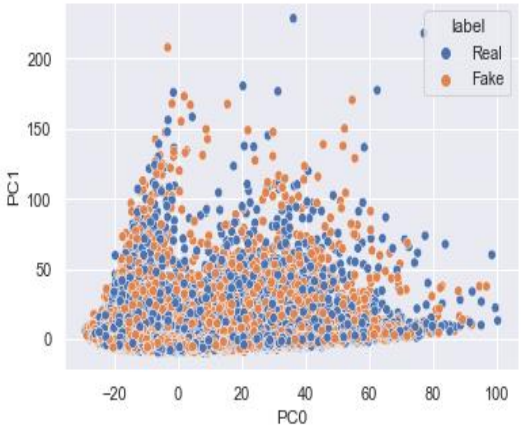
VGGFace



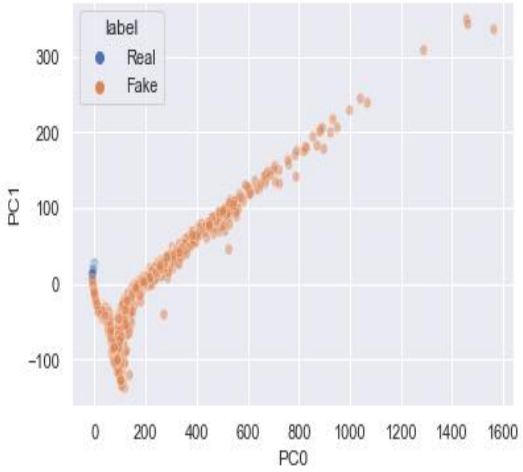
Densenet



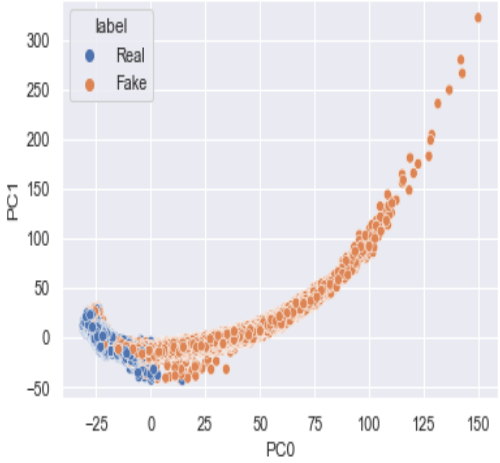
Densenet with Aug Data



Densenet with Graysacle data



ResNet101



InceptionV3 with Aug Data

Conclusion

Deepfake is a very serious and intriguing problem in this age. More and more technologies are discovered every day to tackle the situation. But still, even when the accuracy of models is very high it makes up quite the percentage in the plethora of images that are available in this day and age on social media like Google and Facebook. Similar to the medical domain a slight mistake in the detection can be fatal to the image and reputation of a person. GANs with their technology have only made matters worse.

Our models have shown how well the pre trained models can do on the fake image detection along with our custom model. Even though the models can be 99% accurate, the models were trained with the help of supervised learning and the performance of models on never seen before GANs remain an intriguing questions. Future work could include using autoencoders or using the unsupervised learning with cluster formations.

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.

[2] https://www.kaggle.com/xhlulu/140k-real-and-fake-faces, .

[3] https://www.kaggle.com/ciplab/real-and-fake-face-detection, .

[4] T. T. Nguyen, C. M. Nguyen, D. T. Nguyen, D. T. Nguyen, and S. Nahavandi. Deep learning for deepfakes creation and detection. arXiv:1909.11573, 2019.

[5] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision, pages 2223–2232, 2017.

[6] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adver- sarial networks, 2018.

[7] https://www.reddit.com/r/fakeappreddit/.

[8] https://www.reddit.com/r/sfwdeepfakes/comments/8a2sj1/openfaceswap_an_actual_deepfakes_gui/.

[9] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking, 2018.

[10] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts, 2018.

[11] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses, 2018.

[12] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting World Leaders Against Deep Fakes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. IEEE, June 2019.

[13] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2016.

[14] Tai Do Nhu, In Na, and S.H. Kim. Forensics face detection from gans using convolutional neural network, 2018.

[15] Yuezun Li, Xin Yang, Baoyuan Wu, and Siwei Lyu. Hiding faces in plain sight: Disrupting ai face synthesis with adversarial perturbations, 2019.

[16] Deep Residual Learning for Image Recognition: Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun

[17] Rethinking the Inception Architecture for Computer Vision: Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna