World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:15, No:6, 2021

Generative Adversarial Network Based Fingerprint Anti-Spoofing Limitations

Yehjune Heo

***Abstract*—**Fingerprint Anti-Spoofing approaches have been actively developed and applied in real-world applications. One of the main problems for Fingerprint Anti-Spoofing is not robust to unseen samples, especially in real-world scenarios. A possible solution will be to generate artificial, but realistic fingerprint samples and use them for training in order to achieve good generalization. This paper contains experimental and comparative results with currently popular GAN based methods and uses realistic synthesis of fingerprints in training in order to increase the performance. Among various GAN models, the most popular StyleGAN is used for the experiments. The CNN models were first trained with the dataset that did not contain generated fake images and the accuracy along with the mean average error rate were recorded. Then, the fake generated images (fake images of live fingerprints and fake images of spoof fingerprints) were each combined with the original images (real images of live fingerprints and real images of spoof fingerprints), and various CNN models were trained. The best performances for each CNN model, trained with the dataset of generated fake images and each time the accuracy and the mean average error rate, were recorded. We observe that current GAN based approaches need significant improvements for the Anti-Spoofing performance, although the overall quality of the synthesized fingerprints seems to be reasonable. We include the analysis of this performance degradation, especially with a small number of samples. In addition, we suggest several approaches towards improved generalization with a small number of samples, by focusing on what GAN based approaches should learn and should not learn.

***Keywords*—**Anti-spoofing, CNN, fingerprint recognition, GAN.

I. Introduction

FINGERPRINT recognition is a strong and useful biometric system because of the uniqueness of fingerprints. It can be used as a secure and easy way to identify humans. However, spoof fingerprints violate the secureness of fingerprint recognition systems. For fingerprint recognition systems to be more secure and widely implemented, the systems need to have liveness detection schemes that can detect spoof fingerprints from the live ones. One of the methods in liveness detection techniques in hardware [1] is the detection of blood pressure. Atsushi et al. [2] demonstrated the correlation between live fingerprints and the blood movement inside the skin layer. The problem with these techniques is that it is expensive to be used widely. In order to provide secure and economical fingerprint recognition systems, the use of Convolutional Neural Networks (CNN) is widely investigated. It is possible to train CNN in order to distinguish between fake and live fingerprints. Researchers have been trying to find the most accurate CNN that can distinguish between live and spoof fingerprints, see [17]-[19] for various attempts on detecting spoof fingerprints. Eunsoo et al. [3] demonstrated a CNN for fingerprint liveness detection using the Gram module. The model had an error rate of 2.61%, which shows the potential of using CNN for fingerprint liveness detection.

A Style-Based Generator Architecture for Generative Adversarial Network (StyleGAN) [20] is used in this paper to generate fake images of both live and spoof fingerprints. For the CNN models, AlexNet [12], VGGNet [14], ResNet [15], and DenseNet [16] are used. The CNN models are trained with only the real dataset first, and then the CNN models are trained with the real dataset with added generated-fake images. The accuracy and mean average error rate are recorded for each model to compare the CNNs' performance based on the addition of Style-generated fake fingerprints. Evaluating the performance of StyleGAN on the LivDet dataset by comparing the accuracy of the CNN models is the main purpose of this paper.

The paper is organized as follows. Section II contains the necessary background information. Section II *A* includes the necessity of detecting spoof fingerprints, and Section II *B* explains the difference between live and spoof fingerprints, and Section II *C* types of CNN that will be used in this experiment. The basics and structure of StyleGAN are explained in Section III with the architecture and description. In Section IV, the experiments will be demonstrated by generating the fake images using StyleGAN and observing and comparing the accuracy of each CNN model. The results will be in Section V and the conclusion will be followed in Section VI.

II. Background

A. Spoofing Attacks

There have been many attacks by using fake fingerprints, see [4]-[8] for various spoof attempts. Philip et al. [8] demonstrated an algorithm, DeepMasterPrints, that used deep learning to create a spoof fingerprint and tested it on a database. The spoof fingerprint matched 78% of the real one in the lowest security level of 1%. They also said that "the underlying method is likely to have broad applications in fingerprint security as well as fingerprint synthesis." The use of deep learning to make high-quality spoof fingerprints will lead to a more serious problem to solve.

Recently, there was a case where famous smartphones have been easily unlocked by using just silicone phone cases [6]. These problems indicate that still, many fingerprint recognition systems have difficulty in classifying between live and spoof fingerprints, although there have been significant

Yehjune Heo is with the Korea Internationl School, Yongin-si, Gyeonggi-do 16918 Republic of Korea (phone: 010-7696-3573; e-mail: yehjune111@gmail.com).

improvements in this research.

Implementing liveness detection algorithms into the systems is a probable method only for systems that require strong security. In order to implement fingerprint recognition for everyone to use in important cases, like verifying a bank account, the system needs to be more reliable and cheaper to implement. The use of CNN models can be a reliable method in distinguishing spoof and live fingerprints, and it is also cheaper to implement than hardware anti-spoofing systems.

B. Difference between Live and Spoof Fingerprints

The difference between live and spoof fingerprints can be found from the scanned images from the sensor.

(a) (b)

Fig. 1 Image of a spoof fingerprint made out of gelatine (a) and a live fingerprint (b)

Fig. 1 shows the difference between spoof (a) and live (b) fingerprints. Live fingerprints, when contacted by the sensor, show natural pattern distribution over the fingerprint and natural movements. Live fingerprint images also show the distribution of pores which are small to be made by spoof fingerprints. The image of a live fingerprint in Fig. 1 shows the overall gray region, specific ridges, and distribution of pores. Spoof fingerprints, on the other hand, show unnaturally distributed patterns, dark on some regions, and abnormal ridges. Spoof fingerprints also show unnatural boundary, white or black blob, and abnormally projected histogram. The image of a live fingerprint in Fig. 1 shows random white and black regions, white spots at the edge, abnormal ridges, and no pores. Pores are too small to be found in spoof fingerprints, so it is a distinct feature from live and spoof fingerprints. Pores can be used in high-quality sensors to classify between spoof and live fingerprints. Anil et al. [9] demonstrated a fingerprint recognition algorithm using the detection of pores. This algorithm can be useful in detecting spoof and live fingerprints. However, the difference between live and spoof fingerprints are unrecognizable in many current fingerprint recognition systems. There needs to be a more efficient way to distinguish live and spoof fingerprints.

C. Types of CNN

CNN is a deep learning neural network that is usually used in image classification. In 1988 Kunihiko et al. [10] were the first to propose a hierarchy of neural networks capable of visual pattern recognition. The paper shows the basic structure of DNN (Deep Neural Networks) and the use of neocognitron as a universal pattern-recognizer. In 1998, LeCun et al. [11] represented LeNet, which held the basic architecture of CNN in 1998. LeNet contains the basic convolution layers and pooling layers, and it also uses backpropagation.

There are various types of CNN that can be used for classification purposes. Krizhevsky et al. [12] demonstrated AlexNet, a neural network that achieved first place in the ILSVRC (ImageNet Large Scale Visual Recognition Challenge) [13] with the lowest error rate. AlexNet was not the first CNN, but it was the first useful CNN architecture, and the dropout technique used by AlexNet is the basis for every CNN used in the present days.

Karen et al. [14] demonstrated VGGNet in ILSVRC 2014 and achieved first place in the localization task. VGGNet is an improved version of AlexNet, that uses large kernel filters followed by multiple 3x3 kernels. However, ResNet developers, Kaiming et al. [15], suggested a graph about gradient vanishing in VGGNet, a degradation problem. As the network depth increases, accuracy gets saturated and then degrades rapidly. The solution they suggested was to make shortcuts for the gradients. The shortcut connections turn the network into its counterpart residual version allowing the layers to directly use the data when the input and output are the same dimensions. DenseNet is an improved version of ResNet, and it was demonstrated by Huang et al. [16]. DenseNet has shortcuts for every layer so all of the layers are connected. DenseNet is very similar to ResNet except for an equation that sums the input instead of concatenating it, which leads to a substantially different behavior.

The types of CNN introduced above are the CNNs that had high performance in the ILSVRC. They are all useful CNNs for classification, however, they all have different architectures. To classify between spoof and live fingerprints, the CNN that has the highest accuracy in fingerprint databases needs to be investigated.

III. StyleGAN

Terro et al. demonstrated StyleGAN [20], a style-based generative adversarial network that showed the state-of-the-art performance (low FID score) on both of the Celeba-HQ and FFHQ datasets. StyleGAN uses an advanced technique, style-transfer, for the new generator. StyleGAN views images as distinct parts of styles (gender, facial appearance, etc) and the generator uses those styles to create an image.

Fig. 2 shows that compared to the previous generative adversarial networks, StyleGAN's latent z passes through a mapping network composed of fully connected layers. The latent z vector is converted to an intermediate latent vector w, and the w vector is used to generate style-based images from constant tensor. The use of a mapping network allows the data to easily map with the intermediate latent space and also makes the latent vector w to easily control visual attributes.

Through the synthesis network, the 1024×1024×3 images are generated by the transformation of 4×4×512 constant tensor through the convolution and upsampling layers. After each of the convolution layers the Adaptive Instance Normalization is used to adapt style and the style vector, y, is achieved by the affine transformation of vector w. StyleGAN also uses random

noises in each layer of the synthesis network to improve the realisticness of the image.

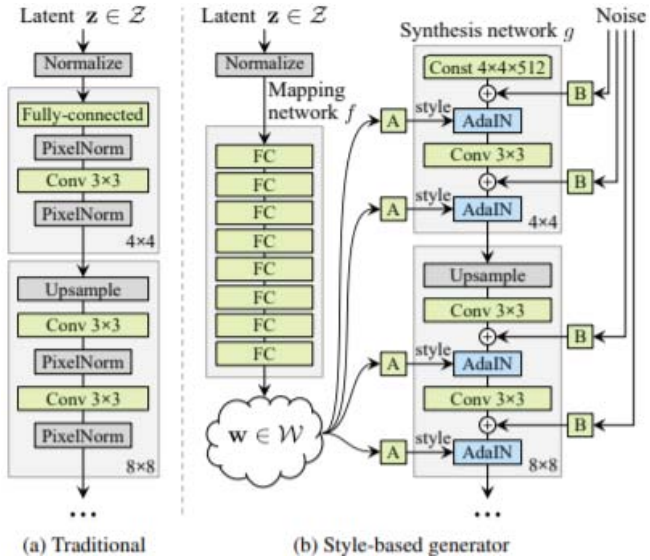

Fig. 2 Image from [20]. Summary of the architecture of StyleGAN

StyleGAN uses the style-mixing technique to resolve the problem of style correlation. Style correlation occurs because the intermediate vector w is used to study all the styles of each layer in the synthesis network. Style-mixing uses two intermediate vectors, w1 and w2, which w1vector is used in the front parts and w2 is used after a specific layer. It is randomly determined when the style is changed for a specific layer, which prevents style correlation from consecutive layers.

IV. Method

The fingerprint database is obtained from the Livdet 2017 database [22]. The models in this paper use 1000 images of each live and spoof fingerprints from the datasets from 2009 to 2017.

(a) (b)

Fig. 3 Examples of images of real live fingerprints in the dataset (a) and examples of images of real spoof fingerprints in the dataset (b)

A. Generating Fake Images Using StyleGAN

This paper used the LivDet dataset to generate 960 fake images for both the live and spoof fingerprint images from StyleGAN [20]. Because of the insufficient number of images, the fake images were generated by multiple data augmented images of the LivDet dataset. This resulted in 7000 images of both live and spoof fingerprints to use for generating fake images. The code for StyleGAN [21] was acquired through Github and the training process ran for 4 days. The generated images are 256×256 and occupy 256.1 KB for each image. For my experiment, StyleGAN was not adjusted, so the quality of the generation of the images were based on the performance of the models.

B. Using CNN Models to Evaluate

The quality of the generated fake images was tested by Convolution Neural Networks. For this, the CNN models were first trained with the dataset that did not contain StyleGAN-generated fake images and the accuracy along with the mean average error rate were recorded. After that, the StyleGAN-generated spoof fingerprint images were combined with real spoof images from the Livdet database, and the StyleGAN-generated live fingerprint images were combined with real live fingerprint images, making each data consist 8000 images. This dataset will be used to train each model.

For CNN's, AlexNet has 26 layers, no data augmentation, pool size of (2,2), kernel size of (3,3), average pooling type, dropout rate of 0.5, zero-padding of (1,1), and activation functions of relu and softmax. VGGNet has 22 layers, no data augmentation, pool size of (2,2), kernel size of (2,2), max-pooling type, dropout rate of 0.5, zero-padding of (1,1), and activation functions of relu and softmax. ResNet uses no data augmentation, pool size of (2,2), kernel size of (3,3), max-pooling type, dropout rate of 0.5, zero-padding of (1,1), and activation functions of softmax. DenseNet uses no data augmentation, pool size of pool size of (2,2), kernel size of (3,3), max pooling type, dropout rate of 0.5, zero-padding of (1,1), and activation functions of softmax.

The CNN models were trained with the dataset of generated fake images and each time the accuracy and the mean average error rate were recorded.

V. Result

The accuracy and the mean average error rate of both are compared to draw a conclusion about the performance of StyleGAN on the fingerprint dataset.

Fig. 4 Examples of images of StyleGAN-generated fake live fingerprints in the dataset

Fig. 4 shows the performance of StyleGAN when generating fake images of live fingerprints. The style-based generation

effectively transfers styles from distinct features in human fingerprints to generate fake fingerprints.

Fig. 5 Examples of images of StyleGAN-generated fake spoof fingerprints in the dataset

Fig. 5 shows the performance of StyleGAN when generating fake images of spoof fingerprints. The style-based generating method effectively generates the flaws in the spoof fingerprint.

For both generated images, the high performance of StyleGAN makes the generated images indistinguishable from the real images.

TABLE I
ACCURACY AND MAE FOR DATASET EXCLUDING STLYEGAN-GENERATED FAKE IMAGES

CNN	Accuracy	MAE	Parameter
AlexNet	92.97%±0.84%	0.07	2,271,194
VGGNet	91.47%±0.64%	0.09	19,110,162
ResNet	92.17%±0.74%	0.07	11,180,674
DenseNet	91.52%±0.77%	0.08	1,074,362

Table of the accuracy and the mean average error rate (MAE) of the CNN models that trained with the dataset that did not contain StylGAN-generated fake images.

Table I shows an average accuracy of 92.03% and a mean average error (MAE) rate of 0.078. Overall, the CNN models perform well with high accuracy and low mean average error rate when training with the dataset that does not contain StyleGAN-generated fake images.

TABLE II
ACCURACY AND MAE FOR DATASET INCLUDING STLYEGAN-GENERATED FAKE IMAGES

CNN	Accuracy	MAE	Parameter
AlexNet	85.20%±1.74%	0.07	2,271,194
VGGNet	80.79%±1.34%	0.09	19,110,162
ResNet	83.46%±0.95%	0.07	11,180,674
DenseNet	85.19%±1.07%	0.08	1,074,362

Table of the accuracy and the mean average error rate (MAE) of the CNN models that trained with the dataset that contained StylGAN-generated fake images.

Table II shows an average accuracy of 83.66% and a mean average error rate of 0.19. Overall, the CNN models do not perform well with low accuracy and high mean average error rate when training with the dataset that contains StyleGAN-generated fake images.

Overall, AlexNet performs high on the dataset with StyleGAN-generated images and relatively high on the dataset without StyleGAN-generated images. 2,271,194 parameters were trained for AlexNet, which is comparably low than the other models. AlexNet performed an average accuracy of 92.97%, which is the highest in this experiment and with a mean average error rate of 0.07 on the dataset with StyleGAN-generated images. For both datasets with and without StyleGAN-generated images, AlexNet seems to be the most suitable model to train on. VGGNet, ResNet, and DenseNet all performed relatively high on the dataset without StyleGAN-generated images, but all failed to perform high on the dataset with StyleGAN-generated images.

When the StyleGAN-generated fake images are combined with the original dataset, there is a significant reduction of accuracy averaging about 8.37% along with an increase of mean average error rate of about 0.112 compared to when the fake images are not combined.

VI. CONCLUSION

This paper studies the limitations of GAN based fingerprint anti-spoofing. When using the original dataset, the CNN models show an average accuracy of 92.03% with an average mean error rate of 0.078. When the StyleGAN-generated fake images are combined with the original dataset, the CNN models show an average accuracy of 83.66% with an average mean error rate of 0.19. The results show that there is a significant reduction of accuracy averaging about 8.37% along with an increase of mean average error rate of about 0.112 compared to when the fake images are not combined.

Although StyleGAN performs high, the results show that GAN based anti-spoofing methods are still vulnerable. Further research is needed to increase the generalization power of the CNN models.

REFERENCES

[1] T. van der Putte and J. Keuning, “Biometrical Fingerprint Recognition: Don’t Get Your Fingers Burned” in *Smart Card Research and Advanced Applications*, pp. 289-306, 2000.

[2] A. Hori and I. Fujieda, “Study on Blood Movement During Fingerprint Input Actions”, *International Journal of Optomechatronics*, Vol. 2, pp.390-400, 2008.

[3] E. Park, X. Cui, W. Kim, and Haki. Kim, “End-to-End Fingerprints Liveness Detection using Convolutional Networks with Gram module”, *ArXiv*, 2018.

[4] E. Marasco and A. Ross, “A Survey on Antispoofing Schemes for Fingerprint Recognition Systems” in *ACM Comput. Surv.*, Vol. 27, pp. 28:1-28:36, 2014.

[5] Arstechnica, “Brazilian docs fool biometrics scanners with bag full of fake fingers”, Available at: https://arstechnica.com, 2013. Accessed on: 9 June 2020.

[6] Arstechnica, “Anyone can fingerprint unlock a Galaxy S10--just grab a clear phone case”, Available at: https://arstechnica.com, 2019. Accessed on: 9 June 2020.

[7] T. Matsumoto, H. Matsumoto, K. Tamada, and S. Hoshino, “Impact of artificial “gummy” fingers on fingerprint systems”, in *Proceedings of SPIE - The International Society for Optical Engineering*, Vol. 4677, 2002.

[8] P. Bontrager, A. Roy, J. Togelius, N. Memon and A. Ross, "DeepMasterPrints: Generating MasterPrints for Dictionary Attacks via Latent Variable Evolution" in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1-9, 2018.

[9] A. Jain, Y. Chen, and M. Demirkus,” Pores and Ridges: Fingerprint

Matching Using Level 3 Features" in *18th International Conference on Pattern Recognition (ICPR'06)*, pp. 477-480, 2006.
[10] K. Fukushima, "Neocognitron A Hierarchical Neural Network Capable of Visual Pattern Recognition", *Neural Networks*, Vol. 1, pp. 119-130, 1988.
[11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition" in *Proceedings of the IEEE.*, Vol. 86, pp. 2278-2324, 1998.
[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Learning Convolutional Neural Networks" in *Advances in neural information processing systems 25(2)*, 2012.
[13] ImageNet, "ImageNet Large Scale Visual Recognition Challenge", Available at: http://www.image-net.org/challenges/LSVRC/, 2015. Accessed on: 9 June 2020.
[14] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks For Large-Scale Image Recognition", *ArXiv*, 2014.
[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition" in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.
[16] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks" in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261-2269, 2016.
[17] E. Park, W. Kim, Q. Li, J. Lim, and H. Kim, "Fingerprint Liveness Detection Using CNN Features of Random Sample Patches" in *2016 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1-4, 2016.
[18] R. F. Nogueira, R. de A. Lotufo, and R. C. Machado, "Fingerprint Liveness Detection Using Convolutional Neural Networks". in *IEEE Transactions on Information Forensics and Security*, Vol.11, No.6, pp. 1206-1213, 2016.
[19] D. Uliyan, S. Sadeghi, and H. A. Jalab, "Anti-spoofing method for fingerprint recognition using patch based deep learning machine" in *Engineering Science and Technology, an International Journal*, 2019.
[20] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks" in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4396-4405, 2019.
[21] Github, "StyleGAN - Official TensorFlow Implementation", Available at: https://github.com/NVlabs/stylegan, 2015. Accessed on: 27 Oct 2020.
[22] LivDet, "LivDet Databases", Available at: livdet.org, Accessed on 2009.