Fooling and Improving Machine Learning based Fingerprint Recognition Systems with Generative Adversarial Networks

Yuxuan Zhou
Syracuse University
yzhou168@syr.edu

Kai Li
Syracuse University
kli111@syr.edu

Novelty:

- We proposed new attack against machine learning based fingerprint recognition systems by leveraging GAN to generate synthetic fingerprints.
- We analyzed the quality of synthetics fingerprint by Poincare algorithm and found that the GAN is effective to generate synthetic fingerprints, and the synthetic data-set follows a similar distribution of Poincare patterns as the real data-set.
- We evaluated existing machine learning based fingerprint recognition systems with synthetic fingerprints, the evaluation result shows the systems are vulnerable and can falsely accept synthetic fingerprints.

Individual contribution:

- Yuxuan Zhou generated synthetic fingerprints with an GAN framework, evaluated a machine learning based fingerprint recognition system with synthetic fingerprints, and drafted the manuscript.
- Kai Li proposed the idea of the paper, evaluated the quality of synthetic fingerprint with the Poincare algorithm, and drafted the manuscript.

Potential venues: NeurIPS 2022, BTAS 2022, etc.

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Abstract

Fingerprint recognition is used for personal identity verification in many places such as smartphones, smart homes, and border customs services. With the advanced development of machine learning methods, state-of-the-art recognition systems have adopted machine learning models to to improve the performance of fingerprint recognition, which achieve a high accuracy compared to conventional fingerprint recognition systems. However, existing research demonstrated that machine learning models are vulnerable to adversatial exmaples and Generative Adversarial Networks(GAN) is such a powerful tool to generate adversarial examples. In this paper, we use GAN to generate synthetic fingerprints and evaluate whether existing machine learning based fingerprint recognition systems shows robustness against synthetic fingerprints. The GAN model uses the same fingerprint training data-set as the machine learning recognition model. Our evaluation result on the synthetic fingerprints shows that GAN is effective to create synthetic fingerprints in a high quality, and synthetic fingerprints follow a similar distribution of Poincare unique patters as the real fingerprints in training data-set. In addition, the evaluation result on the machine learning recognition systems shows that the system can falsely accept more than 64.5% fake fingerprints at a confidence of 0.9.

1 Introduction

Biometrics can be divided into two categories: behavioral features, which represent a specific action patterns determined by humans' body type, lifetime experience, and habits, such examples including signature and walking Gait; another category is biometric features, which are humans' innate characteristics, such as fingerprint, iris, and DNA. Existing work have revolved around recognition and categorization of biometrics including, but not limited to, fingerprints, faces, palmprints, and iris patterns(1; 2; 3; 4; 5). Fingerprint is one of the most widely used biometrics features for identification over the past 50 years(6). Such an biometric feature has been applied in many aspects of people's daily life, including forensics and policing enforcement, transaction authentication, electrical devices authentication, etc. The equipment used to collect human's fingerprints are being developed towards miniaturization, intelligence, and integration(7). In addition, with the advanced development of machine learning, fingerprint recognition systems have evolved a lot and state-of-the-art machine learning based fingerprint recognition systems can achieve a higher recognition success rate. However, although machine-learning based systems have shown promising performance improvement on the fingerprint recognition, it also brings new challenges. One challenge inherited with machine learining based approaches is that the machine learning model is vulnerable to adversarial examples. Existing research has proposed effective and efficient methods to generate adversarial examples that can fool and attack machine learning based classifiers (8; 9). Generative adversarial networks (GAN) is one of the powerful tools which can create adversarial examples and has demonstrated successful attacks on image classification, textual entailment, and machine translation (10; 11).

To the best of our knowledge, no work has explored GAN to attack machine learning based fingerprint recognition systems, and whether state-of-the-art machine learning based fingerprint recognition systems are robust against the adversarial examples generated by GAN remains unknown. To answer this question, we take the first step to apply GAN to generate synthetic fingerprint images and evaluate the robustness of existing machine learning based fingerprint recognition systems against synthetic fingerprint images. Toward the goal, we leverage an existing GAN framework to generate fake fingerprint images and train the GAN framework over a public real fingerprint data-set. Then we feed synthetic fingerprint images into an existing machine learning based fingerprint classifier and test whether the classifier will falsely accept the fake fingerprint. We select the SOCOFing (12) fingerprint data-set that contains 6000 fingerprint images as the training data-set to the GAN framework. We configured the GAN framework to generate more than 3360 fake images. After generating the synthetic fingerprint images, we applied the Poincare algorithm to measure the quality of the generated synthetic images. Finally, we evaluate the robustness of an existing Convolutional Neural Network (CNN) based recognition system against these synthetic images.

The contribution made by our work is summarized as follows.
New attack: our work proposed a new attack against existing machine learning based fingerprint

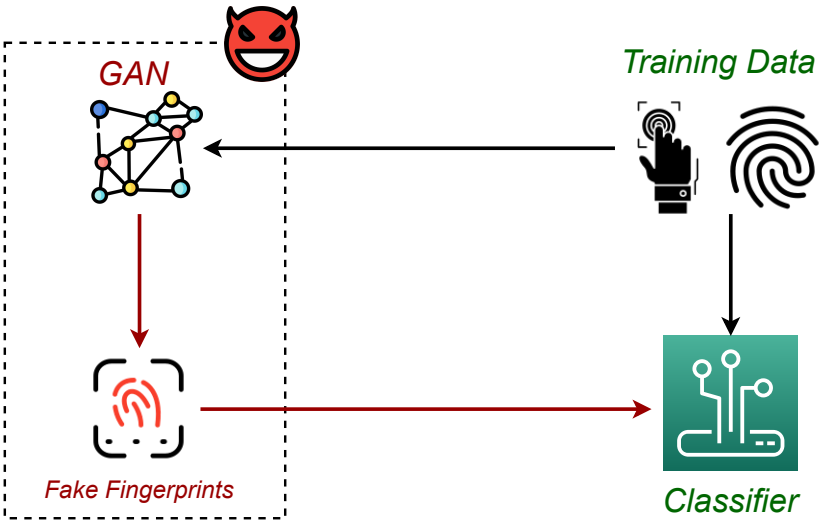


Figure 1: Threat model

recognition systems. The attack works by training GAN over the training data-set used by the recognition system and generating fake fingerprint to fool the system, which allows an attacker to pass the authentication and obtain illegal privilege.

New result of the quality of GAN: we applied an existing GAN framework to generate fake fingerprints and evaluated existing fingerprint recognition systems by fake fingerprints. Our measurement result on the fake fingerprints show that the GAN framework is able to generate fake fingerprint in a high quality, and the generated fake fingerprints images follow a similar distribution of unique Poincare patterns.

New result of the robustness of recognition systems: we evaluated an existing CNN based fingerprint recognition systems with synthetic fingerprints generated by the GAN and measured its robustness. The result shows the CNN model can be easily attacked, with more than 64.5% fake fingerprints can be accepted by the model at a confidence of 0.9.

Roadmap: The remainder of this paper consists of the following. We introduce the preliminary of our work in section 2. The threat model and our proposed attack is presented in section 3. Section 4 illustrates the evaluation results of our work. The novelty and limitation of our work is discussed in section 5. Section 6 concludes our paper.

2 Preliminary

Generative Adversarial Networks(GAN): GAN provides a way to learn deep representations without extensively annotated training data, which are achieved by deriving back-propagation signals through a competitive process involving a pair of networks[11]. GAN can be used in image synthesis and classification, image style transfer, and so on. The function of image synthesis can be applied to generate like-real fingerprints based on feed data. Therefore, GAN is a novel method to generate synthetic fingerprints to test the robustness of existing fingerprint recognition algorithms.

SOCOFing Dataset: The SOCOFing (12) is a biometric fingerprint database designed for academic research purposes. SOCOFing contains 6,000 fingerprint images from 600 African subjects. There are 10 fingerprints per subject and all subjects are 18 years or older. SOCOFing contains unique attributes such as labels for gender, hand and finger name as well as synthetically altered versions with three different levels of alteration for obliteration, central rotation, and z-cut.

3 Threat Model and Attack

We consider a threat model presented in Figure 1. Our threat model involves two actors, namely the attacker and the classifier, and one fingerprint training data-set. First, there is a neural network based

fingerprint classifier that has been trained over the training dataset. Second, there is an attacker who is armed with state-of-the-art generative adversarial networks and is able to access the training data-set. In the presented threat model, the attacker's goal is to use GAN to generate fake fingerprint images and fool the classifier. The attacker first downloads the training data-set used by the classifier and then trains her GAN model. After training, the attacker is able to generate fake fingerprint images. The attacker presents a fake fingerprint image to the classifier, through which the attacker can pass the authentication and obtain illegal access privilege. One may challenge our threat model as follows. Since the attacker can access the training data-set, why could the attacker not present a real fingerprint image to pass the Classifier. However, in an authentication system, we argue that there would be some other auxiliary measure to prevent an attacker from impersonating a real fingerprint in the training data-set (e.g., through IP address or photo ID). Therefore, the attacker can only present fingerprints that are not included in the training data-set.

4 Evaluation

In this section, we first describe how we set up the experiment environment and generate the synthetic fingerprint data-set. Then we illustrate the method we used to measure the quality of synthetic data-set and present the measurement result. Finally, we evaluate different machine-learning fingerprint classifiers with the synthetic data-set and summarize the evaluation result.

4.1 Experiment Setup

Environment: The whole GAN fingerprint generation systems, qualification part and fingerprint recognition systems are implemented in Python and depending on packages including numpy, pytorch, tenserflow, keras, matplotlib, sklearn, imgaug. Our experiment is conducted on a computer equipped with AMD Ryzen 3700x processor@3.6GHz, 32GB RAM, and NVIDIA RTX 3080 graph card.

Generating synthetic data-set: We choose the public fingerprint data-set, SOCOFing, as the training data-set for our GAN. We leverage an open-sourced GAN fingerprint generator (13) to create synthetic fingerprints. To accelerate the training process, the generator altered each fingerprint images from the original 96×103 to 64×64. The program creates a new discriminator network representing the biometric authorization system which is trained on the original data-set. Then, a new generator network is created which uses the weights and paramters captured during the training process. After setup, the generator will produce new fingerprints from a random input vector, display them, and display the output from the discriminator for that fingerprint. The discriminator only uses the original data-set for training, representing a group of known fingerprints for authorized users. The generator takes as input a random vector and produces several batches of new fingerprints, which are then checked against the discriminator. After running the generator, any synthetic images which have confidence $> 90\%$ will be deemed as a good synthetic fingerprint image and saved into the synthetic data-set. In total, we generated 3360 synthetic fingerprint images, among which, the discriminator selects 166 fake finerprint images.

Machine learning fingerprint recognition system: We choose an open-sourced CNN based fingerprint classifier (14) as our evaluation target. Figure 2 presents the architecture and workflow of the classifier. The classifier takes two fingerprint images as input. For each image, the classifier convolve and pooling the image consecutively to get their identity feature model. After that, the classifier will subtract two images' futures and then apply the convolution and pooling again. In the last step, the classifier flattens the matrix and condenses it twice to 1 dimension layer, then uses the binary cross-entropy to output the confidence of two images' similarity.

4.2 Quality of Synthetic Data-set

To measure the quality of the synthetic data-set, for fingerprint images in the real data-set and synthetic data-set, we apply the Poincare detection algorithm to extract unique patterns contained in each image and count the appearances of each pattern. After that, we plot the distribution of different patterns detected by the algorithm across both the real data-set and synthetic data-set, and compare the result. By comparing the distribution of unique patters detected in the real data-set and synthetic data-set, it can give us an overview of the quality of the synthetic fingerprints. For example, if two

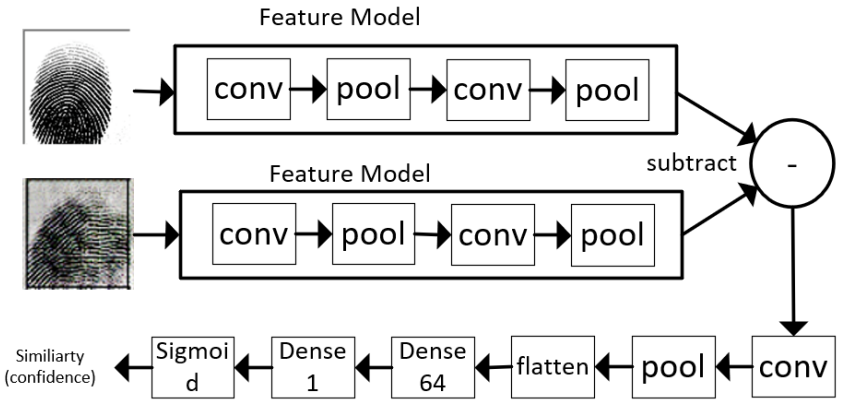
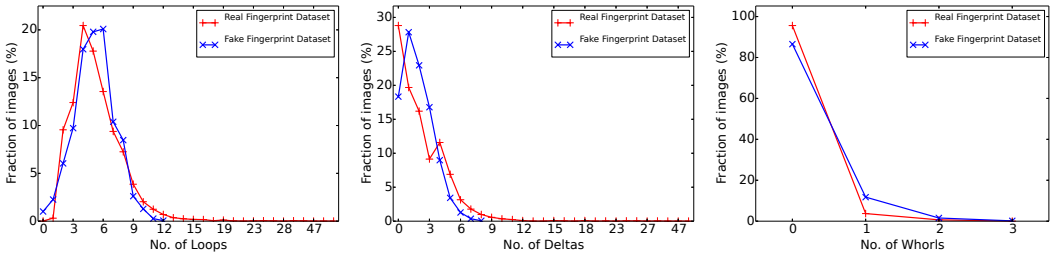


Figure 2: The architecture of CNN based recognition system

data-sets follow a similar distribution of each pattern, we deem the synthetic data-set showing a good quality, otherwise, it has a poor quality.



(a) Detected Loops (b) Detected Deltas (c) Detected Whorls

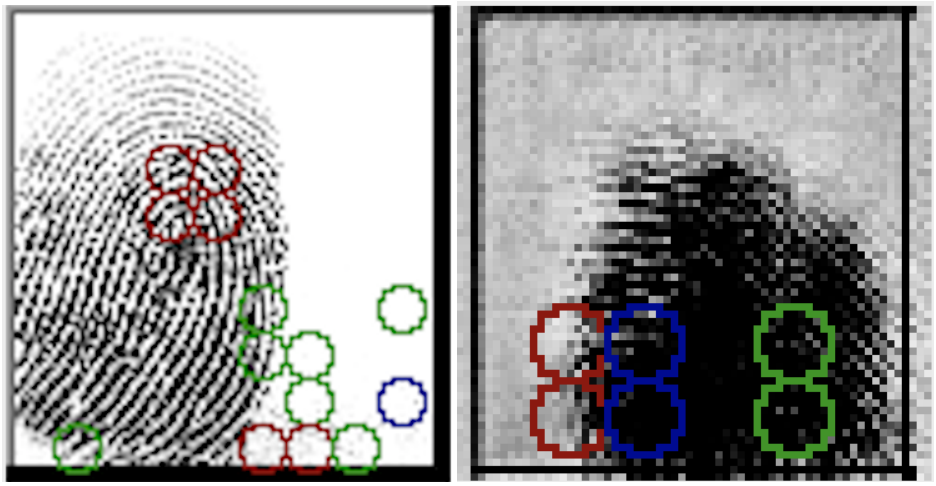
Figure 3: Comparison of real data-set and synthetic data-set

Poincare algorithm: The algorithm works by first dividing each image into blocks of specified block_size. Then for each block, it calculates the orientation of each fingerprint ridge in the block (i.e. what is the rigde slope/angle between a ridge and horizon). By summing up the differences of angles (orientations) of the surrounding blocks, the block will be classified into 4 cases: 1) Loop, if the sum is 180; 2) Delta, if the sum is -180; 3) Whorl, if the sum is 360; 4) None, otherwise.

Poincare analysis result: From the real data-set and synthetic data-set, we respectively select one image to analyze with the Poincare algorithm and plot the detected patterns. The detected patterns are marked with circles in different colors as presented in Figure 4, with red circles showing detected Loops, and green circles showing Deltas, and blue circles showing Whorls.

The statistics of different patterns are described as follows.

- **Loop:** The percentage of images that contains different number of Loops in two data-sets is presented in Figure 3a. From the real data-set, the number of Loops detected by the Poincare algorithm ranges from 0 to 50, with an average of 5.4. In the synthetic data-set, the number of Loops detected by the algorithm ranges from 0 to 12, with an average of 5.2. From the figure, it can be seen that two data-sets follow a similar distribution, with most of the fingerprint images have 3 to 9 Loops, and more than 20% of images have 6 Loops.
- **Delta:** The percentage of images that contains different number of Deltas in two data-sets is presented in Figure 3b. From the real data-set, the number of Deltas detected by the algorithm ranges from 0 to 49, with an average of 2.2. In the synthetic data-set, the number of Deltas detected by the algorithm ranges from 0 to 8, with an average of 1.9. The figure



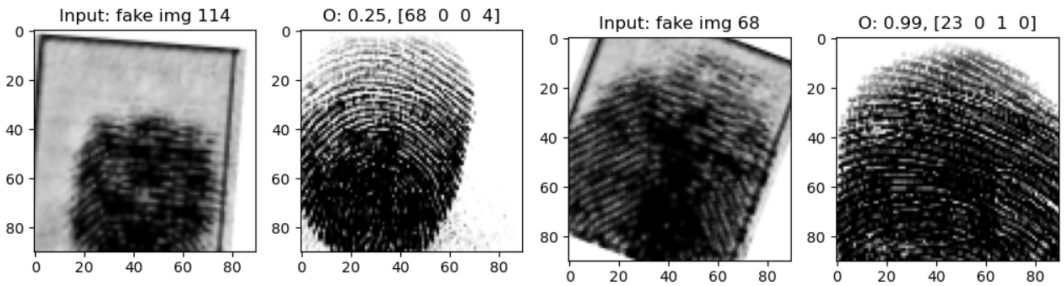
(a) A real fingerprint (b) A synthetic fingerprint

Figure 4: Detected patters by Poincare

shows two data-sets also follow a similar distribution, with most of the fingerprint images have less than 6 Deltas, and more than 25% of images have less than 2 Deltas.

- **Whorl:** The percentage of images that contains different number of Whorls in two data-sets is presented in Figure 3c. From the real data-set, the number of Whorls detected by the algorithm ranges from 0 to 3, with an average of 0.05. In the synthetic data-set, the number of Whorls detected by the algorithm ranges from 0 to 3, with an average of 0.15. The figure shows two data-sets also follow a similar distribution, with most of the fingerprint images have less than 1 Whorl, and more than 80% of images have no Whorls.

Overall, the Poincare analysis results on the real data-set and synthetic data-set show that the distribution of different number of unique Poincare patters in the synthetic data-set resembles to that of the real data-set, implying that the GAN used in our work performed well in generating synthetic fingerprints.



(a) Unmatched fake fingerprint (b) Matched fake fingerprint

Figure 5: Fake fingerprints matching result

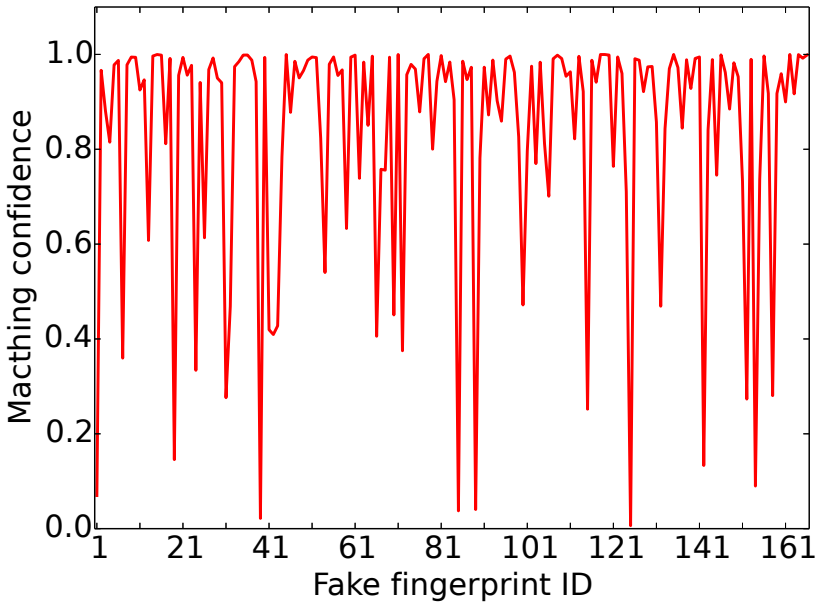


Figure 6: Matching confidence of 166 fake fingerprints

Confidence	Fraction of fake images (%)
>0.9	64.5
0.8 - 0.9	11.4
0.7 - 0.8	7.8
< 0.7	16.3

Table 1: Distribution of matching confidence on fake fingerprints

4.3 Evaluating Machine Learning Recognition Systems

To evaluate the robustness of the selected CNN based fingerprint recognition model, we first train the model with the SOCOFing dataset for 15 epochs. The training process terminated with loss of 0.0440 and accuracy of 0.9840. After training, we have a classifier model that is ready for matching two fingerprints images.

For an input fingerprint image, the classifier will compare the similarity of the input image to each of 6000 real fingerprint images in the training data-set and output the real image that has the highest similarity with the input image. We feed all of 166 fake fingerprint images to the classifier and record all of the output real images and their matching confidence. We select two pairs of fake image and real images that show a high matching confidence and a low matching confidence respectively. Figure 5a shows unlikely a pair of matched fingerprints, the classifier only has a 0.25 confidence to tell they are from the same subject. Figure 5b shows a highly matched fake fingerprint and real fingerprint, and the classifier has a 0.99 confidence to tell they are from the same subject.

We plot all of 166 fake fingerprint images matching confidence over the real fingerprint data-set in Figure 6. The result shows that most of the fake fingerprint images can make the classifier have at least 0.8 confidence to match them to a real fingerprint. We summarize their confidence distribution in table 1. It shows more than 64.5% of fake fingerprints achieved at least 0.9 confidence to match, and only 16.3% for fake fingerprints achieved less than 0.7 confidence.

The evaluation result on the CNN based fingerprint recognition model suggests that the model is vulnerable to fake fingerprint attacks. An attacker can use GAN to generate fake fingerprints to easily fool the model to accept fake fingerprints.

5 Discussion

Our study is the first work that applying GAN to evaluate existing machine learning fingerprint recognition models. Our evaluation result suggested that GAN is effective to generate high quality of fake fingerprints, which can be used to fool the machine learning models. This observation calls for optimization and improvement on the machine learning models. One possible improvement is to train the model by incorporating those fake fingerprint into the training data-set.

In addition, there are some potential improvements on our evaluation process. In our current evaluation section, we only considered one public fingerprint data-set, while there are other public data-set available to use. We would improve the evaluation result by considering those public data-sets. Besides, currently, we only evaluated one machine learning based fingerprint recognition model with the synthetic fingerprints, we noticed that there are other deep learning based recognition models. We would expand our evaluation to cover different models.

6 Conclusion

This work explored GAN to attack machine learning based fingerprint recognition systems. We leveraged an existing GAN framework to generate synthetic fingerprints and evaluated an CNN based fingerprint recognition model. The measurement result on the generated synthetic fingerprints shows GAN can create fake fingerprint in a high quality as evaluated by the Poincare detection algorithm. We evaluated the robustness of an existing CNN based fingerprint recognition system with synthetic fingerprints, the evaluation result suggests that the CNN model can be easily attacked and more than 64.5% fake fingerprints will be accepted by the model at a confidence of 0.9.

References

[1] E. Marasco and A. Ross, "A survey on antispoofing schemes for fingerprint recognition systems," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, pp. 1–36, 2014.

[2] S. Minaee and A. Abdolrashidi, "Highly accurate palmprint recognition using statistical and wavelet features," in *2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE)*. IEEE, 2015, pp. 31–36.

[3] K. W. Bowyer and M. J. Burge, *Handbook of iris recognition*. Springer, 2016.

[4] C. Ding and D. Tao, "Robust face recognition via multimodal deep face representation," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 2049–2058, 2015.

[5] S. Minaee, A. Abdolrashidi, and Y. Wang, "Face recognition using scattering convolutional network," in *2017 IEEE signal processing in medicine and biology symposium (SPMB)*. IEEE, 2017, pp. 1–6.

[6] A. K. Jain, K. Nandakumar, and A. Ross, "50 years of biometric research: Accomplishments, challenges, and opportunities," *Pattern recognition letters*, vol. 79, pp. 80–105, 2016.

[7] X. Xia and L. O'Gorman, "Innovations in fingerprint capture devices," *Pattern Recognition*, vol. 36, no. 2, pp. 361–369, 2003.

[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015.

[9] M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang, "Generating natural language adversarial examples," 2018.

[10] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ser. IJCAI'18. AAAI Press, 2018, p. 3905–3911.

[11] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," 2018.

[12] Y. I. Shehu, A. Ruiz-Garcia, V. Palade, and A. James, "Sokoto coventry fingerprint dataset," 2018.

[13] S. Longofono, "Fingergan," https://github.com/SLongofono/FingerGAN.

[14] T. B. Lee, "fingerprint_recognition," https://github.com/kairess/fingerprint_recognition.