Excel@FIT 2023
http://excel.fit.vutbr.cz

Fingerprint Identity Preserving Generative Adversarial Networks

Bc. Ján Kačur*

Abstract
This project focuses on generating fingerprints using Generative adversarial networks. The special emphasis is put on generating multiple latent fingerprints given the clean fingerprint, with the same identity. The identity and the style should be controllable separately. The chosen approach is based on AugNet model. In the generator, style adaptation from StyleGAN is used. Binarized version of MOLF DB1 dataset is used as clean fingerprint dataset. For latent fingerprint datasets, a subset of NIST SD302, as well as MOLF DB4 are utilized. Trained model is able to generate 3-5 visually distinct latent fingerprint styles per one clean fingerprint in both datasets. The original ridges are visible in the generated fingerprints.

*xkacur04@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

With growing popularity of neural networks in multiple fields of research and industry, fingerprint identification and matching systems have been also leveraging these models. To train such models, one needs large fingerprint datasets. However, due to privacy concerns, many of previously available datasets have been redacted from public use [1]. In addition, there is even smaller amount of public latent fingerprint datasets, as is outlined in Table 1 ([2], [3], [4], [5], [6], [7]).

To combat this issue, this work is focused on generating latent fingerprint images. The ideal features of such system are: generating latent fingerprints from clean ones, preserving identity of the clean fingerprints, generating multiple impressions of one clean fingerprint, and being able to control fingerprint identity and style separately.

When it comes specifically to generating latent fingerprints from clean ones, there are 2 state-of-the-art methods worth mentioning. The first one of them [8] uses CycleGAN [9] model to translate between clean and latent fingerprint domains. The advantage of this method is that it does not need to have 1-to-1 mapping between clean and binarized fingerprints. However, it is hard to generate multiple impressions per one clean fingerprint, as this is only achieved via fine-tuning the CycleGAN models on a specific clusters of latent fingerprints, which becomes intractable for small datasets.

The second approach called AugNet [10] tries to train a generator that can generate latent fingerprint given clean binarized fingerprint, and a latent vector that represents the style and distortions found in latent fingerprints. It only requires a small scale supervised dataset of pairs of clean and latent fingerprints with the same identity.

The method used in this work is based on AugNet approach, and it uses generator inspired by StyleGAN [11] model. The style based generator is combined together with U-Net architecture model used in AugNet, to adapt the latent fingerprint style in identity-invariant way.

2. Latent fingerprint generation

2.1 The objective

Figure 1 in poster shows the objective of this work. We aim to generate latent fingerprints from the clean fingerprints, using conditional generator trained in adversarial manner. The identity of generated fingerprints should be preserved, and the style and distortions are represented with a vector from some latent space. All fingerprint images are grayscale, and they are 512x512 pixels in size.

2.2 First training stage

For the first training stage, shown in Figure 2, we need 2 datasets: dataset of binarized clean fingerprints, and dataset of latent fingerprints. They do not need to be matched via identity. In this stage, we train conditional GAN [12] model, using generator and discriminator. The loss function used for the GAN is LSGAN [13], and the discriminator follows PatchGAN [14] architecture, where the output of the last convoluational layer of size 32x32 is averaged to provide final discriminator output. The latent vector is sampled from normal distribution, and has a dimensionality of 16.

2.3 Generator architecture

The authors of the paper introducing AugNet provided the generator architecture specifics, including the U-Net architecture and the sizes of individual layers. However, they did not provide information on how the latent vector is applied onto the fingerprint. For that, I chose the approach inspired by StyleGAN [11] model. I used the mapping network described in the paper to map the input noise vector to intermediate latent space. That latent space vector is then applied to the convolution feature maps only in the upsampling part, using fully connected layers and AdaIN blocks. The sigmoid is applied as the last activation function to limit the pixel brightness to an interval of $(0; 1)$. Figure 5 describes the generator architecture. KxCySz describes a convolutional layer with kernel size x, y channels, and with a stride of z. Upsampling part contains fractional strides (1/2). All of the convolutional layers except the last one are followed by instance normalization and Leaky ReLU. FCx represents fully connected layer with output size of x.

2.4 Second training stage

In the second training stage depicted in Figure 3, we use the trained generator from the first stage to serve as a dataset generator. The encoder network tries to approximate the sampled latent vector from the image, that was generated using this vector.

2.5 Third training stage

In the third training stage in Figure 4, another dataset is needed. This dataset should contain pairs of binarized clean and latent fingerprints, with the same identity. The trained encoder from the second stage extracts the latent vector from the latent fingerprint. This vector is constrained by KL divergence. The vector is passed to the generator, along with the clean fingerprint. The image produced by the generator should look realistic (hence Discriminator 2), and it should look as similar as possible to the original image (hence L1 loss). The final objective is the weighted sum of all losses mentioned in all steps.

3. Experiments & results

All training stages and data pipelines have been implemented. However, due to the instability of GAN models, most of the focus so far has been directed towards the first (and partially second) training stage. If the first training stage is not able to produce reasonably realistic fingerprints, then the other 2 stages will be unstable. As mentioned before, for the GAN in the first stage, LSGAN [13] loss is used, as it provided better results than traditional BCE, which led to vanishing gradients in the generator. The clean fingerprints used for the network were adopted from MOLF DB1 [3] and binarized (4000 images in total). For the latent fingerprint datasets, both the subset of NIST SD302 [2], and the MOLF DB4 [3] datasets were used. Examples of generated images are shown in Figure 6 and Figure 7. Some of the generated images contain substantial amount of noise. In addition, the network is suffering from mode collapse, as it fails to generate more than 3-5 visually different fingerprint styles for one clean fingerprint. This also makes it difficult to train the encoder in the second stage, as one image style may correspond to potentially large number of different latent vectors. I also experimented with joining the first and the second training stages (as is described in the original AugNet paper [10]) to reduce mode collapse, but currently to no avail. I also tried using multiple discriminators for individual subsets of the latent fingerprints dataset, which also proved to be unsuccessful. On the flipside, the generated fingerprints seem to have similar ridge structure to the original image, which seem to be a sign of identity preservation.

4. Conclusion

This paper is based on my diploma thesis currently in progress. Of the 3 training stages, most work has been done on the first one, due to instability issues, which would then make another 2 stages also unstable. The model can produce 3-5 visually different latent fingerprint styles from 1 clean fingerprint, with the quality high enough, that the original ridges can be seen. The biggest problem is the mode collapse, which needs to be addressed further. Furthermore, proper evaluation methods need to be designed to assess the image quality, and the identity preservation.

References

[1] Joshua J. Engelsma, Steven A. Grosz, and Anil K. Jain. Printsgan: Synthetic fingerprint generator. *CoRR*, abs/2201.03674, 2022.

[2] Gregory P Fiumara, Patricia A Flanagan, John D Grantham, Kenneth Ko, Karen Marshall, Matthew Schwarz, Elham Tabassi, Bryan Woodgate, Christopher Boehnen, et al. Nist special database 302: Nail to nail fingerprint challenge. 2019.

[3] Anush Sankaran, Mayank Vatsa, and Richa Singh. Multisensor optical and latent fingerprint database. *IEEE Access*, 3:653–665, 2015.

[4] Anush Sankaran, Tejas I. Dhamecha, Mayank Vatsa, and Richa Singh. On matching latent to latent fingerprints. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–6, 2011.

[5] Anush Sankaran, Akshay Agarwal, Rohit Keshari, Soumyadeep Ghosh, Anjali Sharma, Mayank Vatsa, and Richa Singh. Latent fingerprint from multiple surfaces: Database and quality analysis. In *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, 2015.

[6] Michael Garris and R. McCabe. Nist special database 27 fingerprint minutiae from latent and matching tenprint images, 2000-06-01 2000.

[7] Anush Sankaran, Mayank Vatsa, and Richa Singh. Hierarchical fusion for matching simultaneous latent fingerprint. In *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 377–382, 2012.

[8] Andre Brasil Vieira Wyzykowski and Anil K. Jain. Synthetic latent fingerprint generator, 2022.

[9] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.

[10] Ying Xu, Yi Wang, Jiajun Liang, and Yong Jiang. Augmentation data synthesis via gans: Boosting latent fingerprint reconstruction. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2932–2936, 2020.

[11] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018.

[12] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[13] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, and Zhen Wang. Multi-class generative adversarial networks with the L2 loss function. *CoRR*, abs/1611.04076, 2016.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.