Biometrics in the Age of Generative AI

Rishabh Shukla

Department of Computer Science Engineering
Indian Institute of Technology, Jammu, India
ORCID (Rishabh Shukla): https://orcid.org/0009-0008-6818-4674

Abstract.

With the rise in the popularity of artificial intelligence, generative methods in biometric has garnered significant attention. While several current techniques are capable of generating and restoring biometric traits such as faces and fingerprints while maintaining the original identity, despite their ability to provide information, there is still a large gap between real and synthetic data. We are working on a framework that is capable of disentangling and addressing these tasks. Our approach involves the use of a Convolutional-based Network architecture(CNN, GAN, DDPM, and Hybrid models). These approaches provide enhanced quality and authenticity, but because of their intricacy, they can pose challenges in processing and are less adaptable. We are evaluating our approach against the existing SOTA methods and also performing an ablation study.

1 Methodology

We can divide our proposed work into two objectives:

1. **Objective 1:** Face generation and restoration
2. **Objective 2:** fingerprint generation and restoration.

The primary contributions of our study are as follows:

1. Our proposed methodology represents a disentangled and subject-agnostic approach. This method tackles the challenging task of generating and restoring features.
2. We develop a novel network architecture based on the UNET approach to enhance its capability in handling these tasks.
3. In our knowledge, we are first working on large missing parts for all biometric modalities.

The general framework of proposed approach is revolving around key based generation. When provided with a task of generation or to fill in a given image, we determine the nearest cluster corresponding to its skin tone or class. Next, we use the key to generate or fill specific unique identity of the given image. Using this information, we can accurately determine the information such as ethnicity, expression, and gender of the given image. Ultimately, we get our output with enhanced precision and lifelike characteristics as a result of the key-specific model. The general steps can be summarized as:

For Generation

A. Intermediate Image Generation($M_N \in 0,1^{200*136*1} \rightarrow I^{200*136*1}$): In this step, we design and train variational autoencoders $VAE_{M_N \in 0,1^{200*136*1}}$ to take the input noise matrix $M_N \in 0,1^{200*136*1}$ and generate an intermediate image $I^{200*136*1}$. This transformation majorly imparts semantic characteristics of a fingerprint template. Here we use a variational autoencoder ($VAE_{M_N \rightarrow I}$) to generate the intermediate image from the noise matrix $M_N \in 0,1^{200*136*1}$.

B. Generating unique fingerprint identity from intermediate image($I^{200*136*1} \rightarrow T^{400*256}$): This step uses the intermediate image ($I^{200*136*1}$) generated in the previous step and transforms it in latent representations (LV_I). LV_I is projected to new randomized mapping in the target class domain according to the class-specific key K. The randomized latent representation(LV_I') is then transformed using a trained generator model to generate a unique fingerprint identity $T^{400*256}$.

C. Generating multiple impressions from unique fingerprint identity ($T_I^{400*256} \in I_1, I_1, I_1 - -I_N$): Here we utilize an impression generator module $G_{T_I^{400*256} \in I_1, I_1, I_1 - - I_N}$ to generate the multiple impressions of the generated unique-ID (in the previous step). This transformation ensures that the impression must have variation between them but also store characteristics of intra and inter-user.

For Restoration The restored unique identity will be referred to as $R^{256*256}$, while the mask will be indicated as $M_I \in 0,1^{256*256}$. The input distorted image, $I \in 0,1^{256*256*1}$, will be used as the input. The key employed for restoration particular to each class is denoted as $K^{1*256} \in I_1, I_2, I_3, I_4, I_5$. The restoration methodology can be categorized into three components.

A. Mask Learning and finding cluster from Input Image ($M_I \in 0,1^{256*256} \leftarrow I \in 0,1^{256*256*1}$): In this stage, we utilized convolution to handle the input, which is a deformed image represented as $I \in 0,1^{256*256*1}$. The goal was to acquire knowledge of the mask M_I and key, which belongs to the set of binary matrices of dimensions 256x256, based on the given image. This procedure primarily identifies the pixels that are concealed in the original image. This stage will act as a fundamental component for the following step.

B. Using a key to initiate class-specific restoration ($K^{1*256} \in I_1, I_2, I_3, I_4, I_5 \rightarrow U_{I \in 0,1^{256*256*1} \rightarrow R_T \in T_1, T_2, T_3, T_4, T_5}$): This phase utilizes the key K^{1*256}, which is determined by the type of fingerprint or face image submitted. The provided key activated the model to enhance the given image by including more precise characteristics of the corresponding class.

C. Performing inpainting on the specified valid pixels $I \in 0,1^{256*256*1} \rightarrow \mathbf{R}^{256*256}$: In this process, we employ our U-NET module, denoted as $U_{R_I^{256*256} \in I_1, I_2, I_3, I_4, I_5}$, to recover the absent portion of the provided biometric image, represented as $I \in 0,1^{256*256*1}$. This restoration is facilitated by utilizing a key, denoted as K^{1*256}, and a mask, denoted as $M_I \in 0,1^{256*256}$ (specified in the previous step). This restoration guarantees that the images

will have variations between them while still preserving the distinctive properties of class-specific identities.

2 Outcome Observations and Results Obtained Till Now

In order to evaluate the effectiveness of the proposed work, we tested generated samples in terms of EER, AUC, and TAR@FAR. Below are some results and generated samples from the published and unpublished work.

2.1 Quantitative Performance

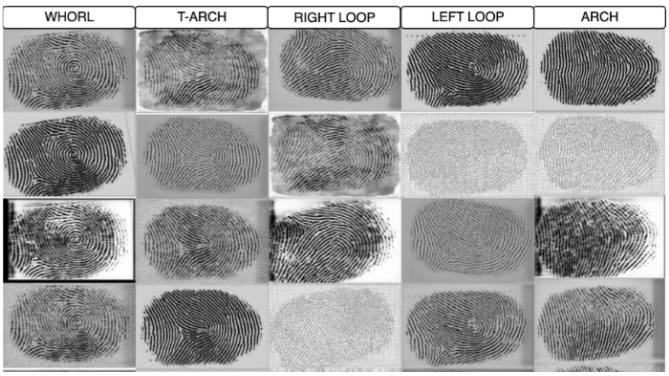

Figure 1. Some generated samples.

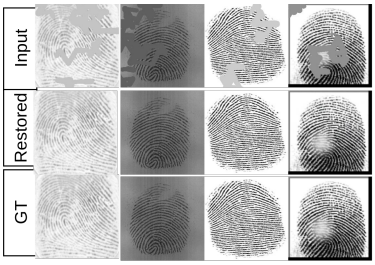

Figure 2. Some restored samples.

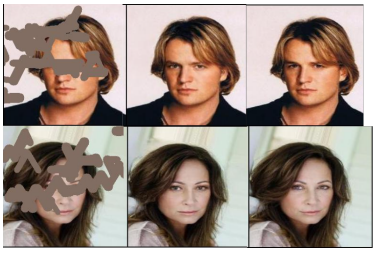

Figure 3. Some restored samples(Left to right: Input, Restored, Ground Truth).

We have tested our work for both. Face and Fingerprint.

2.1.1 Fingerprint

a) Intra-user Scenario: In this case, for anguli and Socfing, we used 3 impressions $I_N \in I_2, I_2, I_3$ of every unique identity I in $rest\_database$ and for FVC we used 2 impressions $I_N \in I_2, I_2$ of every unique identity I in $rest\_database$.
b) Inter User Scenario: This scenario studies the case where any unique user wants to authenticate himself, then it should not matched with any other user.

2.1.2 Face

For the face, we have tested our inpainted faces with their respective ground truth image. The calculated scores are used to asses the performance of the framework. All the outcomes are reported and illustrated in Table and Figure 1,2,3.

Table 1. Comparison of performance between our generated samples(Vikriti-ID) and other publicly available datasets using MCC matcher.

Metric	Database	EER%	AUC
Comparision	Vikriti-ID	0.16%	0.9
	SOCOFING	0.17%	0.89
	FVC 2000(DB1)	0.46	0.58
	FVC 2000(DB2)	0.30%	0.73
	FVC 2000(DB3)	0.47%	0.56
	FVC 2000(DB4)	0.29%	0.74
	FVC 2002(DB1)	0.52%	0.44
	FVC 2002(DB2)	0.41%	0.58
	FVC 2002(DB3)	0.3%	0.75
	FVC 2002(DB4)	0.38%	0.66
Performance	Data leakage	0.02%	0.98
	Trained on generated dataset	0.005%	0.99

Table 2. Comparison of performance between OURS restored data with their respective datasets for Fingerprints.

Database	Original		Restored	
	EER%	AUC	EER%	AUC
ANGULI	0.023%	0.98	0.016%	0.99
SOCOFING	0.17%	0.89	0.12%	0.92
FVC 2000(DB1)	0.46%	0.58	0.20%	0.89
FVC 2000(DB2)	0.30%	0.73	0.15%	0.87
FVC 2000(DB3)	0.47%	0.56	0.21%	0.83
FVC 2000(DB4)	0.29%	0.74	0.15%	0.91
FVC 2002(DB1)	0.52%	0.44	0.32%	0.65
FVC 2002(DB2)	0.41%	0.58	0.42%	0.54
FVC 2002(DB3)	0.3%	0.75	0.10%	0.96
FVC 2002(DB4)	0.38%	0.66	0.35%	0.41

Table 3. Image Quality Score comparison of restored face samples with SOTA for Quality Metrics. (Red=Best, Blue=second best)

Metric	**CTSD**	**RN**	**GC**	**PC**	**OURS**
FSIM↑	0.9634	0.9578	0.9670	0.9650	0.9701
HaarPSI↑	0.9018	0.8990	0.9037	0.9120	0.9116
MS-SSIM↑	0.9680	0.9679	0.9720	0.9706	0.9747
VIFp↑	0.7734	0.7850	0.7888	0.7870	0.7897
DSS↑	0.7090	0.7225	0.7150	0.7140	0.7215
VSI↑	0.9921	0.9930	0.9910	0.9952	0.9932
SR-SIM↑	0.9832	0.9887	0.9850	0.9920	0.9917
BRISQUE↓	25.6607	24.8148	25.1000	25.9200	24.5148
ContentLoss↓	325.9589	325.6734	325.1200	325.9000	324.8244
DISTS↓	0.0362	0.0382	0.0357	0.0351	0.0323
PieAPP↓	0.3091	0.3100	0.2832	0.3060	0.2991
GMSD↓	0.0627	0.0625	0.0617	0.0622	0.0608
MDSI↓	0.2682	0.2694	0.2650	0.2673	0.2604
MS-GMSDc↓	0.0690	0.0684	0.0630	0.0673	0.0664
TV↓	28.0753	28.2203	28.9400	28.1200	27.9738