Signature Generation Using GAN

Muhammad Abdur Rafey
Computer Science
FAST NUCES Islamabad
Islamabad, Pakistan
i21070@nu.edu.pk

***Abstract*—This paper presents a novel approach to signature generation using Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs). We explore the complementary strengths of these two models in producing high-quality, realistic signatures. The VAE minimizes a combination of reconstruction loss and KL divergence, enabling effective latent space representation and sample generation. Simultaneously, the GAN employs an adversarial training mechanism, where the Generator creates synthetic signatures while the Discriminator assesses their authenticity. Our findings demonstrate significant improvements in the quality of generated samples, evaluated through qualitative and quantitative metrics. Future work will focus on hyperparameter optimization and the implementation of advanced evaluation methods to enhance the effectiveness of the generated signatures.**

I. Introduction

Signatures are widely utilized for authentication in various domains, including banking, legal documentation, and personal identification. The ability to generate realistic signatures can significantly impact industries by facilitating automated signature verification and enhancing security protocols. Traditional methods for signature generation often fall short in capturing the intricacies and variations inherent in human signatures.

In recent years, deep learning techniques, particularly Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs), have shown remarkable promise in generating realistic images. VAEs are powerful generative models that learn a compact representation of the data in a latent space, enabling smooth sampling and high-quality reconstructions. Meanwhile, GANs utilize a unique adversarial training framework that pits two neural networks against each other, fostering the generation of increasingly realistic images.

This paper investigates the integration of VAEs and GANs for signature generation, leveraging their strengths to produce authentic signatures. We detail the architectures, training processes, and results obtained from our experiments. By analyzing the loss values and visual outputs, we evaluate the performance of the models and outline future directions for optimization and evaluation. Our contributions aim to advance the field of automated signature generation, with implications for security and verification technologies.

II. Model Architectures

A. VAE

- *Encoder*: A convolutional neural network (CNN) compresses the input into a latent space.
- *Latent Space*: The latent vector is sampled using the reparameterization trick (sampling from a Gaussian distribution defined by mu and logvar).
- *Decoder*: A transposed CNN reconstructs the image from the latent vector.

B. GAN

- *Generator*: Converts random noise from the latent space into signature-like images using transposed convolution layers.
- *Discriminator*: Distinguishes between real and fake signatures using CNN layers, outputting a probability score.

III. Training Process

A. Variational Autoencoder (VAE) Training

The VAE aims to minimize a combined loss function that consists of two main components: the reconstruction loss and the regularization term. The reconstruction loss, typically measured using Binary Cross-Entropy (BCE), quantifies how well the VAE can reconstruct the original input from the latent representation. The regularization term, known as Kullback-Leibler (KL) divergence, ensures that the learned latent space distribution is close to a standard normal distribution.

The overall loss function can be expressed as:

$$\mathcal{L}_{VAE} = \mathbb{E}_{q(z|x)}\left[\log p(x|z)\right] - D_{KL}(q(z|x)||p(z)) \quad (1)$$

where:

- x is the input data,
- z is the latent variable,
- $q(z|x)$ is the encoder's output (the approximate posterior),
- $p(x|z)$ is the decoder's output (the likelihood of x given z),
- D_{KL} denotes the KL divergence.

By balancing the reconstruction quality and the smoothness in the latent space, the VAE achieves both good sample generation and meaningful latent representations.

B. Generative Adversarial Network (GAN) Training

GAN training involves a unique adversarial process where two neural networks—the Discriminator and the Generator—are trained simultaneously. The Discriminator is tasked with classifying images as real or fake, learning to differentiate between genuine samples from the training set and those produced by the Generator. The Generator, on the other

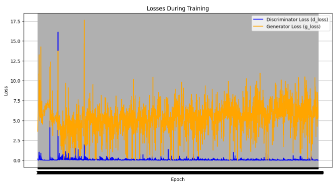


Fig. 1: Generator Loss vs Discriminator Loss

hand, aims to produce realistic images that can deceive the Discriminator.

The training alternates as follows:

1) Train the Discriminator to maximize the probability of correctly classifying real and fake images.
2) Train the Generator to minimize the Discriminator's ability to distinguish between real and generated images.

The objective functions can be summarized as:

$$\mathcal{L}_D = -\mathbb{E}_{x \sim p_{data}}[\log D(x)] - \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))] \quad (2)$$

$$\mathcal{L}_G = -\mathbb{E}_{z \sim p_z}[\log D(G(z))] \quad (3)$$

where:

- $D(x)$ represents the Discriminator's output for real images,
- $G(z)$ represents the images generated by the Generator,
- p_{data} denotes the real data distribution,
- p_z denotes the noise distribution used to generate fake images.

This adversarial training encourages both networks to improve over time, stabilizing their performance as training progresses.

IV. Results

A. Loss Values

The VAE's reconstruction loss decreases steadily across training epochs, indicating improved performance in reconstructing input samples. The GAN's losses for both the Discriminator and Generator reveal a dynamic interplay between the two networks, with the losses stabilizing over time as they converge toward an optimal balance.

B. Signature Generation

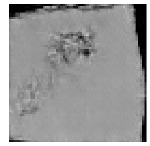
New samples are generated using both the VAE and GAN architectures. These generated images are saved for further evaluation and analysis to assess their quality and authenticity. The result is shown in 2a,2b and 2c.

V. Next Steps

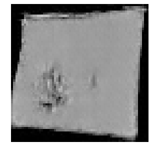
A. Hyperparameter Tuning

To optimize the performance of both models, we will fine-tune several hyperparameters:

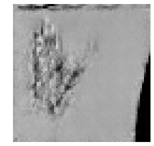
- Learning rates for both the Generator and Discriminator in the GAN,
- Dimensions of the latent space in both models,
- Number of training epochs to ensure adequate training without overfitting.



(a) Generated Signature from Group 0



(b) Generated Signature from Group 103



(c) Generated Signature from Group 11

Fig. 2: Generated signatures from different groups using GAN

B. Evaluation Metrics

To quantitatively evaluate the quality of the generated samples, we will consider incorporating metrics such as:

- **Inception Score (IS)**: Measures the quality of generated images based on the Inception model's predictions.
- **Fréchet Inception Distance (FID)**: Compares the distributions of real and generated images in the feature space.