This is a pre-print of the paper accepted for presentation at the International Conference on Computer Vision and Pattern Recognition (CVPR 2026)

Fingerprint Fragment Expansion using Image Outpainting Approach based on Spectral Normalization PatchGAN

C. Zaghetto•, A. Purim•, W. Oliveira•, J.R. Ribeiro Jr.•, H. Nolla•, F. Santos•, M. Chang°, R.H.Vareto•
•Samsung R&D Institute Brazil | °Samsung Electronics South Korea
{c.zaghetto,a.purim,witor.o,jose.junior,moons.chang,r.vareto}@samsung.com

Abstract

Partial fingerprint authentication systems, such as the ones used in smartphones or tablets, are used for a wide range of tasks. However, the small sensor sizes in these devices limit system performance, creating a trade-off between high false acceptance rates (low security) and high false rejection rates (low usability), or rendering some algorithms unusable. This paper introduces a technique for expanding fingerprint fragments using a Generative Adversarial Network (GAN) model with a patch-based loss function and Spectral Normalization (SN) for weight normalization. The model expands the image by generating ridge lines with local continuity based on neighboring information and global patterns. Our experiments show that expanding fragments with the proposed method enables the reconstruction of minutiae outside the fragment border, reducing the equal error rate (EER) by approximately 0.524% *and* 0.34% *in the NIST Biometric Image Software Suite Bozorth3 and VeriFinger 13.1 SDK matchers, respectively.*

1. Introduction

Fingerprint biometric systems are among the most reliable and common methods for authentication [16]. Devices such as smartphones use fingerprints for a variety of applications, including sensitive tasks like banking. However, the small sensors in these devices result in systems that operate on fingerprint fragments rather than full images. These partial fingerprint authentication systems face limitations and vulnerabilities, such as increased false rejection and acceptance rates (FRR and FAR, respectively), and greater susceptibility to attacks like MasterPrint [15]. Enhancement techniques have been proposed to improve the quality of fingerprint fragments and increase the accuracy and reliability of verification processes. However, these techniques do not address image size limitations, which can render some algorithms unreliable or unusable[1].

[1]The NFIQ2.0 documentation states that input images should be at least 196x196 pixels after cropping any whitespaces from the edges.

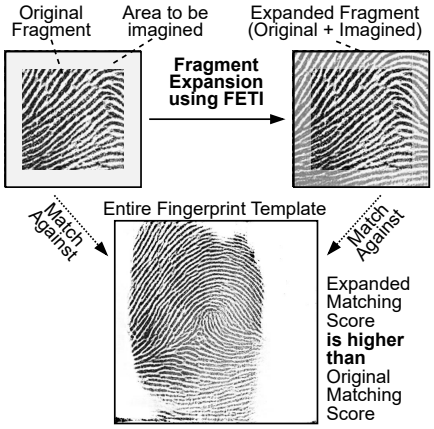

Figure 1. Example of the proposed solution: the original fragment (top left) is expanded by generating an imagined region using a generative model; the expanded fragment (top right) has higher quality and matching scores against the fingerprint template when compared to the scores from the original fragment; the differences in shade of gray are used only to visually distinguish the original fragment from the imagined area.

This paper addresses this size limitation by artificially expanding the spatial dimensions of fingerprint fragments using an outpainting model. Our proposed framework uses a Generative Adversarial Network (GAN) based on the SN-PatchGAN architecture [20], with a two-stage generator comprising a coarse network and a two-stream refinement network, both employing dilated gated convolutions and contextual attention layers. Given an input fingerprint fragment, an expansion mask, and an empty context matrix, the model generates an outpainted fragment by predicting realistic ridge patterns. This synthesis procedure uses local neighborhood information while maintaining global structural coherence. The generated output is evaluated by a patch-wise Markovian Discriminator, which classifies individual fingerprint patches as real or fake to penalize local discontinuities. Spectral Normalization layers stabilize adversarial training by normalizing network weights, ensuring smoother gradients and mitigating mode collapse.

We created a specialized fingerprint expansion model, referred to as Fingerprint Enhancement Through Imagination (FETI), by adapting and fine-tuning a generalist SN-PatchGAN to the fingerprint domain with a diverse dataset of synthetic fingerprints. As demonstrated in our experiments, FETI effectively extrapolates fragments past their original borders, using internal ridge patterns to accurately reconstruct minutiae structures outside the initially known area. To evaluate the proposed solution, *NFIQ 2.0*[1] assesses fingerprint quality before and after expansion. Additionally, *Bozorth3*, *MINDTCT*, and *VeriFinger* compute fingerprint matching scores and FAR/FRR values. The first two are part of the NBIS suite[2], while *VeriFinger* is a commercial SDK[3]. Figure 1 presents a real example of a fingerprint fragment expanded using the FETI model.

The primary contributions of this study are: *(i)* introduce a novel fingerprint enhancement methodology based on fragment expansion via a specialized, fine-tuned SN-PatchGAN; *(ii)* demonstrate that expanded fragments yield higher biometric quality and matching scores than their non-expanded counterparts; *(iii)* systematically evaluate the correlation between fragment expansion size and overall performance metrics; and *(iv)* show that, by expanding fingerprint fragments, it is possible to accurately reconstruct minutiae structures outside the original boundary that directly correspond to the ground-truth image.

2. Related Works

Computer vision techniques to enhance fingerprint images are particularly relevant in partial fingerprint authentication systems, where the system lacks access to a complete fingerprint image.

Since minutiae extraction algorithms depend heavily on input image quality, which can be compromised by artifacts and noise, enhancement is a crucial step to improve reliability [5, 7]. Traditional enhancement techniques focused on separating ridge patterns from noise and used classic image processing methods, such as contrast stretching, histogram manipulation, normalization, and Wiener filtering. These approaches are often referred to as pixel-wise enhancements, where the new value of a pixel depends solely on its original value and a global parameter, without considering neighboring pixels.

To address the shortcomings of pixel-wise enhancements, contextual filtering was introduced. This method differs from other approaches by using domain-specific knowledge, such as ridge orientation and frequency. Proposed by Hong *et al*. [5], the approach consists of: *(i)* energy normalization, *(ii)* ridge orientation estimation, *(iii)* estimating local ridge frequency and *(iv)* applying locally adapted Gabor filters, which utilize local orientation and frequency information. Unlike pixel-wise enhancements, contextual filtering adjusts the filter's characteristics according to the local context. Since its introduction, this method has remained a widely used and effective technique for fingerprint enhancement, with numerous variations documented in the literature [10].

The effectiveness of traditional enhancing methods for fingerprint quality and feature extraction facilitation has been well documented [1, 4, 12, 16]. However, these methods are limited in their ability to reconstruct missing fingerprint regions or create new structures [18], as they primarily focus on noise suppression and ridge pattern enhancement. More recently, Deep Neural Networks have been applied to fingerprint enhancement tasks [2, 13, 17]. While these approaches show promising results, they do not address outpainting techniques. This study focuses on generative models, which are better suited for the fragment expansion task proposed here.

Most generative methods for fingerprint enhancement focus on either denoising or inpainting fingerprint images. For instance, Prabhu *et al*. [14] proposed U-Finger, a method based on a multi-scale convolutional network with dilated convolutions and multi-scale cascaded feature modules. Adiga and Sivaswamy *et al*. [18] came up with FPD-M-Net, utilizing a M-Net-based architecture with a combination of per-pixel and structure similarity loss functions. Liang and Liang designed ResWCAE [9], a deep learning architecture using a Residual Wavelet-Conditioned Convolutional Autoencoder (Res-WCAE) with Kullback-Leibler divergence regularization for biometric image denoising.

GANs have also been explored for fingerprint enhancement: Zhu *et al*. [22] proposed FingerGAN, which employs a skip-connected denoising autoencoder as the generator to translate low-quality latent prints into higher-quality ridge structures. Similarly, Huang *et al*. [6] used a conditional GAN with a U-Net-based generator and a PatchGAN discriminator for latent fingerprint enhancement.

While inpainting techniques have been widely studied for recovering missing regions within fingerprints, outpainting (expanding) fingerprint fragments remains largely unexplored. *To the best of our knowledge, none of the currently existing works have investigated the potential of increasing the spatial extent of fingerprint fragments to improve matching accuracy in partial fingerprint authentication systems.*

3. Proposed Approach

The proposed FETI model for fingerprint outpainting, based on the SN-PatchGAN [20] architecture is capable of expanding fingerprint fragments while preserving the spatial coherence of ridge patterns and minutiae structures.

[1] https://www.nist.gov/services-resources/software/nfiq-2

[2] https://www.neurotechnology.com/verifinger.html

[3] https://nist.gov/services-resources/software/nist-biometric-image-software-nbis

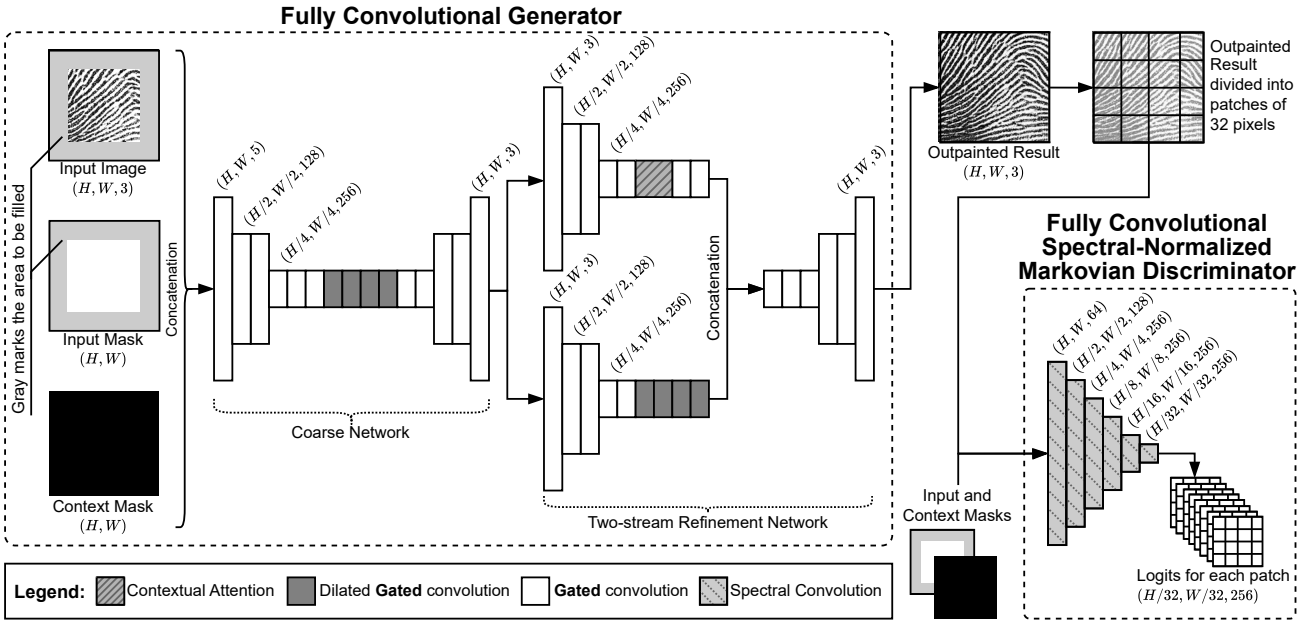

Figure 2. Architecture of the generator network with dilated gated convolution and contextual attention mechanism alongside the discriminator network encompassing Markovian patches and spectral normalization.

3.1. FETI Generator

As shown in Figure 2, the two-stage generator comprises a coarse and a refinement network made up of dilated gated convolution and contextual attention modules. The former encompasses a dilated gating mechanism that controls the flow of information and enhances the model's ability to learn more complex patterns, whereas the latter allows the generator to focus on relevant parts of input feature maps. A detailed description of these modules is provided below.

3.1.1. Dilated Gated Convolution

Dilated convolution, also known as *atrous* convolution [3], is a variant of the conventional convolution that allows for larger receptive fields without increasing the computational cost and the number of parameters. It is effective in tasks such as image reconstruction, where capturing context from a larger surrounding area of the input is important.

For clarity, let $Conv(\mathcal{W}, F, d)$ be a dilated convolution operation applied to a single-channel input sample F and parameters set with a typical singular stride, as follows:

$$\tilde{F} = Conv(\mathcal{W}, F, d)$$

$$\tilde{F}_{[x,y]} = \sum_{i=\lfloor -\frac{K}{2} \rfloor}^{\lfloor \frac{K}{2} \rfloor} \sum_{j=\lfloor -\frac{K}{2} \rfloor}^{\lfloor \frac{K}{2} \rfloor} \mathcal{W}_{[i,j]} \cdot F_{[x+i*d, y+j*d]} + \mathcal{W}_b, \quad \forall\, (x,y) \in F \quad (1)$$

where $\mathcal{W}_{[i,j]} \in \mathbb{R}^{K \times K}$ represents the convolution weight matrix indexed by $[i, j]$, $\mathcal{W}_b \in \mathbb{R}$ stands for the bias value, and $d \in \mathbb{Z}_{\geq 1}$ refers to the dilation factor that sets the spacing between kernel elements. In addition, $F \in \mathbb{R}^{H \times W \times 1}$ designates any tensor on which the convolution is applied to and $\tilde{F}$ indicates the resulting feature map. The double summation iterates over the filter/kernel dimensions, which is typically square, and the $\forall$ notation implies that this operation is performed for every pixel position (x, y) in F.

Gated convolution is a type of convolutional layer that learns a probability mechanism, called *gate*, to regulate the flow of information through the neural network [20]. In this study, we define it as the combination of dilated convolutional functions over a corrupted fingerprint image F and its corresponding mask M in such a way that:

$$\tilde{F} = Gated(F, M, d) = \phi(Conv(\ddot{\mathcal{W}}, F, d)) \odot \sigma(Conv(\breve{\mathcal{W}}, M, d)) \quad (2)$$

where $\ddot{\mathcal{W}}$ corresponds to the filter weights and their joint bias for feature extraction whereas $\breve{\mathcal{W}}$ comprises the kernel weights and bias for gate mapping. Symbol $\odot$ designates an element-wise multiplication between tensors, ϕ stands for any non-linear activation function and σ indicates the *sigmoid* function that maps real-valued numbers into probability scores bounded by 0 and 1. The underlying idea behind gated convolutions is to multiply each output element of a dilated convolutional layer $\phi(Conv(.))$ by its matching learned gating score $\sigma(Conv(.))$ that controls the extent to which the signal passes through. This mechanism allows the network to preserve coherent ridge directions without being misled by uninformative or noisy pixels.

3.1.2. Contextual Attention

Contextual attention uses the information from the nearby context (background) to guide the attention mechanism in the reconstruction of missing regions (foreground) [19]. In essence, it consists of extracting patches from known and unknown regions in favor of computing matching scores between them. This procedure is followed by a *softmax* operation, turning attention scores into probabilities and playing a pivot role during the refinement and deconvolution stages.

Let $F^b \in \mathbb{R}^{H_b \times W_b \times C}$ represent the input feature map from the known (background) area and $F^f \in \mathbb{R}^{H_f \times W_f \times C}$ denote the feature map from the missing part (foreground), previously filled by the coarse network, as indicated in Figure 2. Each contextual attention layer begins by extracting $k \times k$ patches from both F^b and F^f. Subsequently, these patches are flattened into vectors, resulting in their respective two-dimensional embeddings, that is, matrices $P^b \in \mathbb{R}^{(H_b \cdot W_b) \times (C \cdot k^2)}$ and $P^f \in \mathbb{R}^{(H_f \cdot W_f) \times (C \cdot k^2)}$.

Note that reshaping the latent feature space eases the process of computing normalized scalar products among all patches through the cosine similarity function: $\mathcal{S}_{[i,j]} = Cos(P^f_{[i]}, P^b_{[j]}) = \left\langle P^f_{[i]} / \left\| P^f_{[i]} \right\|, P^b_{[j]} / \left\| P^b_{[j]} \right\| \right\rangle$ where $P^f_{[i]}$ designates the i^{th} foreground area and $P^b_{[j]}$ stands for the j^{th} flattened background patch. The estimation of angular correspondence among all known and unknown regions results in $\mathcal{S} \in \mathbb{R}^{(H_f \cdot W_f) \times (H_b \cdot W_b)}$, a matrix where each element stores how related patches $P^f_{[i]}$ and $P^b_{[j]}$ are. Next, the attention module applies the *softmax* activation $\bar{\mathcal{S}}_{[i,j]} = \frac{e^{\mathcal{S}_{[i,j]}/\tau}}{\sum_{k=1}^{(H_b \cdot W_b)} e^{\mathcal{S}_{[i,k]}/\tau}}$ to the rows of $\mathcal{S}$, turning angular similarity scores into $[0, 1]$ probabilities and sharpening the distribution through a low temperature τ.

The extracted background patches are weighted by the probability scores considering $\tilde{P}^f = \bar{\mathcal{S}} \cdot P^b$ where $\tilde{P}^f \in \mathbb{R}^{(H_f \cdot W_f) \times (C \cdot k^2)}$ before being reshaped back into the new feature map $\tilde{F}^f \in \mathbb{R}^{H_f \times W_f \times C}$ (values of overlapping pixels are averaged). The resulting $\tilde{F}^f$ encapsulates the refined features for the foreground region, with each patch being a blend of relevant known parts weighted by the attention scores $\bar{\mathcal{S}}$, effectively replacing the original tensor F^f.

Contextual Attention helps the model reuse semantically similar ridge patterns from unmasked areas, thereby enhancing structural consistency and sharpness across reconstructed fingerprint zones. Later, the refinement network concatenates both context and gated-based streams, initiating the deconvolution stage and reconstructing the masked image, which is afterwards fed into the discriminator.

3.2. FETI Discriminator

The FETI architecture employs a patch-wise Markovian discriminator that analyses individual fingerprint patches rather than the entire sample, allowing the model to capture local patterns and textures more effectively. Moreover, it applies a spectral normalization technique to normalize the weights and stabilize the training procedure, boosting learning convergence and preventing mode collapse.

3.2.1. Spectral Convolution

Spectral normalization is a technique to stabilize adversarial training by normalizing the spectral norm (largest singular value) of weight matrices in the discriminator [11]. The training of GANs are generally unstable due to strong gradients in the discriminator, causing the generator to fail. The normalization controls the discriminator updates by ensuring that its weight matrices have bounded singular values, leading to smoother gradients and more stable training.

Theoretically, given a weight matrix $\mathcal{W}$ from a neural network layer, the spectral norm computation starts with the singular value decomposition of $\mathcal{W} = U \Sigma V^T$ where Σ is the diagonal matrix containing singular values while U and V are orthogonal matrices. The spectral norm $\hat{\Sigma}$ is formally defined as $\hat{\Sigma} = \max_{i \in |\Sigma|} \Sigma_i$, the largest singular value of $\mathcal{W}$, often interpreted as the "stretching power" of a matrix. Thereafter, the normalized weight matrix can be defined as $\hat{\mathcal{W}} = \mathcal{W}/\hat{\Sigma}$, stabilizing the adversarial training and ensuring that $\hat{\mathcal{W}}$ holds a norm of 1. This operation guarantees that the discriminator satisfies the *Lipschitz* continuity, controlling gradient explosion and interrupting model collapse.

Computing algebraic decomposition is computationally expensive, especially for large weight matrices. Instead, we approximate the spectral norm $\hat{\Sigma}$ using power iteration:

$$v \leftarrow W^T u, \quad u \leftarrow W v, \quad \hat{\Sigma} \approx u^T W v$$

where u and v and randomly initialized vectors normalized at each iteration in order to ensure numerical stability. As a result, the normalized weight matrix is computed through $\hat{\mathcal{W}} = \mathcal{W}/\hat{\Sigma} \approx \mathcal{W}/u^T W v$, avoiding the full factorization into singular values and making spectral normalization efficient. During backpropagation, the gradients and weight updates are computed with respect to $\hat{\mathcal{W}}$, keeping the spectral norm with the value of 1 after each update, controlling gradient magnitudes and avoiding instability.

3.2.2. Markovian Patches

The Markovian discriminator [8] is designed to capture local region statistics rather than the entire image at once. FETI assumes that an image's spatial information is locally Markovian, that is, the value of a pixel depends only on its surrounding area. As a result, our approach divides the image into smaller patches of 32×32 pixels and classifies each one independently as either real or fake.

The discriminator consists of a convolutional neural network that slides a window over an image and treats each patch individually. More precisely, let $I \in \mathbb{R}^{H \times W \times C}$ be the input image to the Markovian discriminator D. This image I can be either a non-expanded (original) or outpainted fingerprint fragment. The network divides I into overlap-

ping patches so that $D(I) \rightarrow \mathcal{R}_{H' \times W' \times 256}$ results in a grid of probabilities where each element $\mathcal{R}_{[i,j,c]}$ holds the likelihood that a patch centered at (i, j) containing the semantics from filter c is authentic or synthetic. Ultimately, the discriminator outputs $\mathcal{R}$ so that the chosen patch-wise cost function is able to calculate the dissimilarity between each region and its corresponding ground truth.

3.3. Cost Functions

FETI relies on Hinge loss to optimize generator and discriminator networks, as it produces more stable training and prevents gradient saturation. Loss $\mathcal{L}_G$ calculates dissimilarity between real fingerprint and generated images. Conversely, $\mathcal{L}_D$ measures discriminator error when distinguishing real images x from generated images $G(\hat{x})$ where $\hat{x}$ represents the input image with missing regions. Both cost functions are detailed below:

$$\mathcal{L}_D = \mathbb{E}\left[\max(0, 1 - D(x))\right] + \mathbb{E}\left[\max(0, 1 + D(G(\hat{x}))\right]$$

$$\mathcal{L}_G = -\mathbb{E}\left[D(G(\hat{x}))\right] + \mathbb{E}\left[L_1(G(\hat{x}), x)\right]$$

The discriminator error $\mathcal{L}_D$ is minimized when $D(x) = +1$ for real images and $D(G(\hat{x})) = -1$ for generated images. Essentially, $\mathcal{L}_D$ encourages the discriminator to output values close to $+1$ for real image patches and penalizes deviations. Still, $\mathcal{L}_D$ punishes the discriminator when $D(G(\hat{x}))$ is greater than -1 for generated images.

With a different purpose, the generator loss $\mathcal{L}_G$ drives the generator to maximize $D(G(\hat{x}))$, effectively fooling the discriminator by assigning higher scores ($\approx +1$) to reconstructed images. This adversarial mechanism ensures that the generator produces images with structural coherence, pixel-wise alignment and visual fidelity. Furthermore, the $L_1(.)$ term composes a reconstruction loss, enforcing a pixel-wise Manhattan similarity between the generated fingerprint and its ground truth. Respectively, $\mathcal{L}_G$ is directly tied to $\mathcal{L}_D$ output, creating an adversarial training scenario where the generator develops by iteratively producing more realistic images to spoof the discriminator.

4. Experiments and Results

This section presents the fine-tuning, evaluation protocols, qualitative analysis, quantitative evaluation with matching error rates with the *Bozorth3* and *VeriFinger 13.1 SDK* matchers, and quality metrics using the *NFIQ 2.0* tool.

4.1. Fine-tuning and Datasets

Both Generator and Discriminator were originally pretrained on the *Places2* dataset [21] due to its genericity and size, encompassing 10^8 iterations with 16-sample batches, resulting in approximately 200 epochs. *Adam* optimizers were set with a learning rate of 10^{-4}, beta coefficients of 0.5 and 0.999 for both generator and discriminator networks, as specified by Yu *et al*. [20]. The same pretraining parameters were employed when fine-tuning the FETI model for 100 epochs using an AMD EPYC 7742 CPU and a NVIDIA A100 GPU, which took approximately 57 hours.

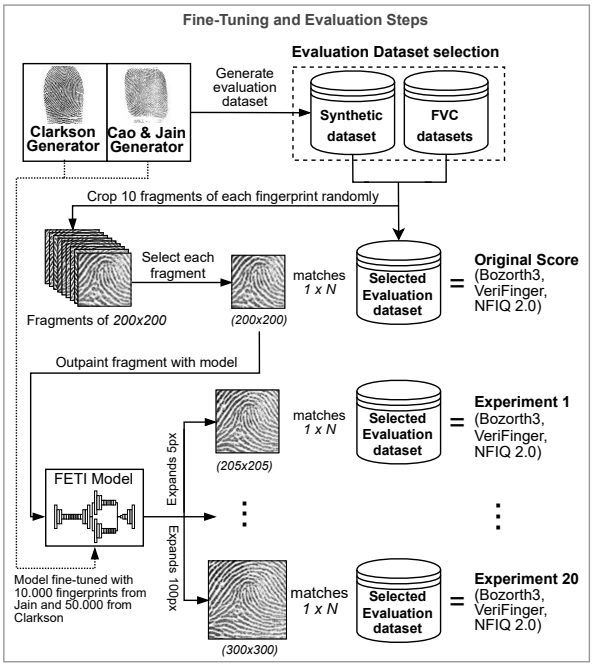

Figure 3. Overview of the fine-tuning and evaluation protocol. The model is evaluated by comparing the expanded fragments against the original fragments to evaluate quality and matching scores.

The methodology for fine-tuning and evaluating the model is illustrated in Figure 3. The fine-tuning process utilized a training set composed of $50,000$ fingerprint templates synthesized using the Clarkson Generator[4] and additional $10,000$ synthetic fingerprints generated via Cao and Jain's approach[5]. At each training batch, random crops were applied to extract fragments holding 416×416, 128×256, or 256×128 pixels, ensuring that both height and width remained multiples of 32.

To evaluate quantitatively, we generated $1,000$ fingerprints with *NFIQ 2.0* score greater than 85 using Cao and Jain's approach, referred to as the **Synthetic dataset**. We also used 960 fingerprints (720 real and 240 synthetic) gathered from the 2000, 2002, and 2004 Fingerprint Verification Competition (FVC), referred to as the **FVC dataset**.

4.2. Evaluation Protocol

To validate the proposed method, we conducted 20 experiments using the Synthetic and FVC datasets. For each experiment, the following steps were executed:

Step 1: Crop fragments. For each fingerprint image in the evaluation datasets, 10 fragments of size 200×200 pix-

[4]https://github.com/keivanB/Clarkson_Finger_Gen
[5]https://github.com/prip-lab/fingerprint-synthesis

els were cropped. This process generated $10,000$ fragments for the Synthetic dataset and $9,600$ for the FVC dataset, referred to as *original* fragments. To avoid mostly blank areas in corners, fragment centers were randomly placed within a radius of 80 pixels from the image center for the Synthetic dataset and 40 pixels for the FVC dataset.

Step 2: Expand fragments. The *original* fragments were outpainted using the FETI model to produce *expanded* fragments. In Experiment 1, fragments were expanded from 200×200 to 205×205 pixels, in Experiment 2, from 200×200 to 210×210 pixels, and so on up to Experiment 20, which expanded fragments to 300×300 pixels. The objective was to determine how far from the original fragment the model could expand with reliable gains in matching scores. This process generated an additional set of $10,000$ and $9,600$ *expanded* fragments for the Synthetic and FVC datasets, respectively, for each of the 20 experiments.

Step 3: Compute quality scores. We measured the fingerprint quality of all the *original* and *expanded* fragments in each experiment using *NFIQ 2.0*.

Step 4: Compute matching scores. We computed matching scores for both the *original* and *expanded* fragments against all fingerprint templates in the evaluation dataset. Two matchers were used: *Bozorth3* with *MINDTCT* as the minutiae extractor, and *VeriFinger 13.1 SDK*. These scores allowed us to evaluate the impact of the expansion method on partial fingerprint authentication systems. Since each fragment was matched against every template, a total of approximately 40 million and 36 million matching scores were computed for the Synthetic and FVC datasets, respectively, in each of the 20 experiments.

Step 5: Compute FAR and FRR curves. We calculated the False Acceptance Rate (FAR) and False Rejection Rate (FRR) curves by varying a threshold. For each matcher (*Bozorth3* and *VeriFinger*), scores at or above the threshold were considered a match (accept), while scores below were considered a non-match (reject). The threshold was varied from zero to the maximum matching score, and the FAR and FRR were calculated at each value. The Equal Error Rate (EER), that is, the point where the FAR and FRR curves intersect, was then measured. To evaluate the impact of the expansion method, we compared the FAR, FRR, and EER values between the *original* and *expanded* fragments.

4.3. Qualitative Minutiae and Texture Analysis

Fine-tuning the FETI model on a diverse fingerprint dataset teaches it to preserve ridge continuity and generate minutiae in expanded regions. These generated minutiae can be either spurious or "reconstructed", respectively lacking a corresponding minutia in the ground truth or correctly matching a minutia present in the ground truth. Figure 4 shows an example where FETI reconstructs a bifurcation minutia in the expanded region. As the model expands further from the known fragment border, the likelihood of generating spurious minutiae increases and the amount of local information of the original fragment decreases.

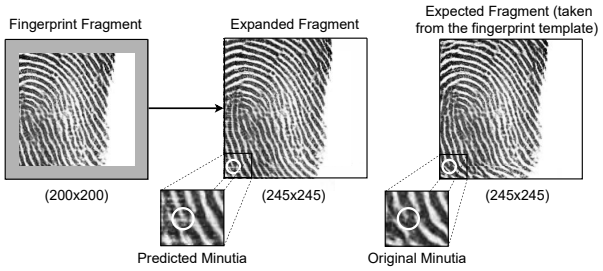

Figure 4. Example of a reconstructed minutiae in an expanded fragment. The generated bifurcation matches a minutiae in the ground truth (expected fragment) within a three-pixel radius.

Our analysis found both spurious and reconstructed minutiae in the expanded outputs, indicating that the model learned to generate minutiae but may occasionally misplace them. We also observed texture inconsistencies in some expanded regions, as the training dataset comprises fingerprints from various sources with different textures.

4.4. Quantitative EER Analysis

To analyze the impact of these observations, we selected *MINDTCT* and *Bozorth3*, popular tools for minutiae extraction and fingerprint matching, respectively. Since both operate on minutiae, they are sensitive to spurious minutiae. We also employed *VeriFinger*, which incorporates texture analysis in matching, to assess whether texture inconsistencies affect performance. To assess the impact of fragment expansion in matching performance, we measured the Equal Error Rate (EER), that is, the point at which the FAR and FRR are equal. Lower EER values indicate more reliable matching. If the expansion method successfully improves fragment quality, a decrease in EER, compared to the original fragments, is expected.

4.4.1. Bozorth3

In the synthetic dataset, the original EER is 0.777% (200×200), while the lowest error rates are 0.181% and 0.172% in Exp. 9 (245×245) and Exp. 19 (295×295), respectively. The average EER for the experiments is 0.253%, indicating a decrease in EER of 0.524%. For the FVC dataset, the original EER is 20.18% and the average EER for the experiments is 18.587%, indicating a decrease in EER of 1.593%. Figure 5 shows a downward trend in EER for the synthetic dataset from Exp. 1 (205×205) to Exp. 8 (240×240), while for the FVC this downward trend is visible from Exp. 2 to Exp. 7. In the FVC dataset, from Exp. 9 (245×245) to Exp. 20 (300×300) an increase in EER is present, but it is still a lower EER than the original (i.e.: Exp. 20 has a EER of 17.63% while the original has 20.18%).

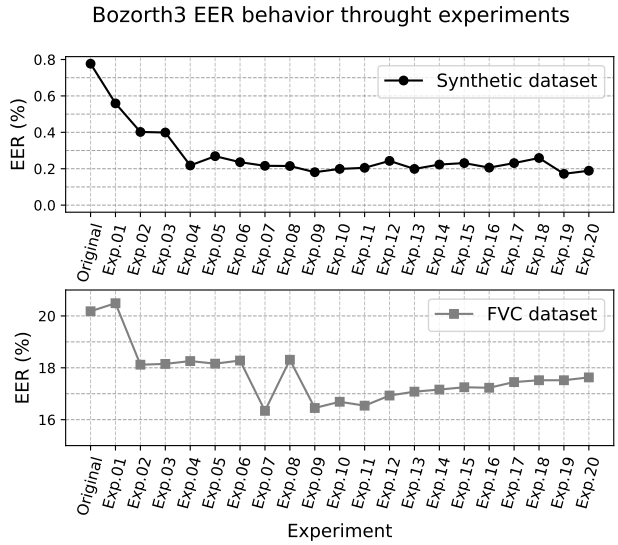

Figure 5. Comparison between the EER computed with the *Bozorth3* tool for the original fragments and each of the 20 expansion experiments made for the Synthetic and FVC datasets.

4.4.2. VeriFinger

As shown in Figure 6, the FAR curves of the synthetic dataset in all experiments are close to the original dataset when evaluated using the *VeriFinger 13.1 SDK* tool, meaning the solution does not significantly increase or decrease the false acceptance rate. In contrast, the FRR curves exhibit changes across different experiments.

The FRR curves of Exp. 3 and Exp. 6 are below the original curve for thresholds from 200 to 350, meaning the model is able to reduce the false rejection. Exp. 9 (245×245) is the first with an FRR curve that stays above the original fragments' FRR curve, and Exp. 18 presents the curve with the higher FRR rates among all experiments. As the FAR curves did not exhibit any significant changes, we can conclude by examining the FRR curves that, when the proposed method is used to expand the *original* fragments to smaller sizes like 215×215 pixels or 230×230 pixels, it can improve the system's performance, as the FRR rates for these *expanded* fragments are smaller compared to the FRR rates of the *original*. However, when using the method to expand the *original* fragments to larger sizes such as 300×300 pixels, a drop in performance may be seen.

In the FVC dataset, Figure 7 shows that the EER for VeriFinger has a decrease from Exp.1 to Exp. 8 and then it starts to increase, even surpassing the Original EER (1.375%) in Exp. 20 (1.443%). The average EER of the experiments is 1.035% indicating an average decrease of 0.34% when compared to the original EER. The increased FRRs and EERs observed in experiments with larger expansions in *VeriFinger* may be attributed to the fact that the model is attempting to generate information in a region that is far from the initial information present in the original fragment, thus decreasing the matching score due to an increase in spurious minutiae and texture inconsistencies.

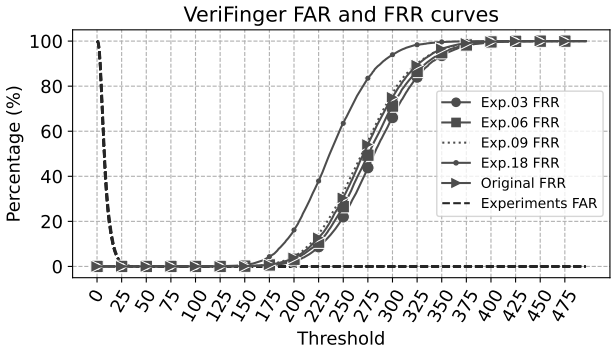

Figure 6. *VeriFinger* FAR and FRR curves for several thresholds used in some of the experiments made with the Synthetic dataset.

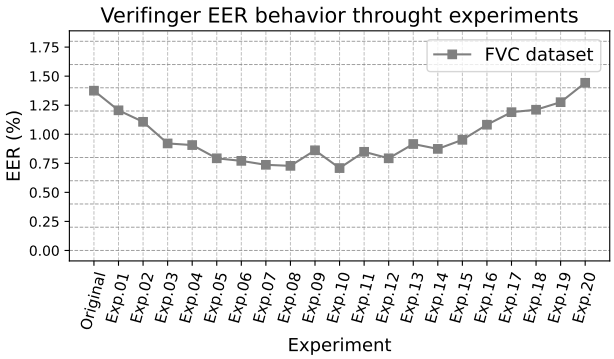

Figure 7. Comparison between the EER computed with the *VeriFinger* tool for the original fragments and each of the 20 expansion experiments made for the FVC datasets.

4.4.3. NFIQ 2.0

Regarding fingerprint quality, our experimental results show that the mean *NFIQ 2.0* score of the expanded fragments is at least as high as the mean score of the original fragments for all experiments. Figure 8 shows the improvement in mean *NFIQ 2.0* score for each experiment, calculated as the difference between the expanded and original fragments. Figure 9 shows the score distributions for the original fragments and the expanded fragments from Exp. 20 for the FVC dataset.

4.5. Next Steps

The model architecture combines multiple components (dilated gated convolutions, contextual attention, spectral normalization, etc.) but their individual contributions remain unclear. An ablation study would help identify which components are essential for fingerprint expansion and whether simpler models could achieve comparable results. Further-

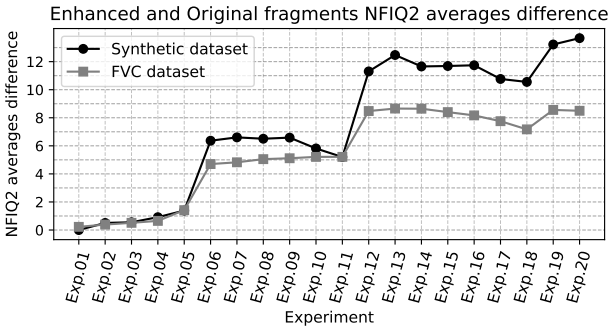

Figure 8. Average *NFIQ 2.0* score difference between the original fragments and the enhanced fragments of all the experiments.

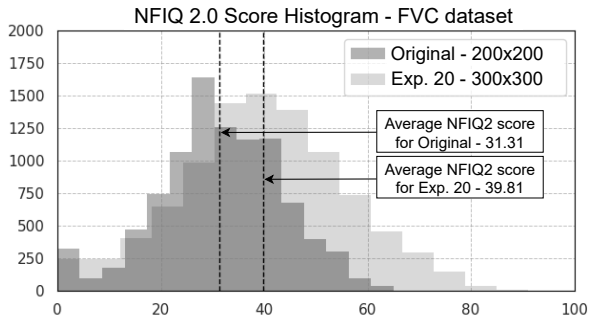

Figure 9. *NFIQ 2.0* score distribution for the original fragments (200×200) and Exp. 20 (fragments expanded to 300×300) for the Synthetic and FVC datasets.

more, comparing the current Generative Adversarial Network against alternative generative architectures, such as diffusion models or variational autoencoders, could reveal whether other paradigms offer advantages in terms of image quality, training stability, or computational efficiency for the fingerprint expansion task.

The current training and evaluation protocol relies predominantly on synthetic fingerprint datasets, complemented by a relatively small set of 720 real samples from the FVC datasets. Expanding the experimental setup to include larger and more diverse datasets composed of both real and synthetic fingerprints would significantly strengthen the validity of the proposed approach. The incorporation of plain, rolled, and latent impressions would expose the model to a wider range of ridge patterns and distortions, thereby improving its robustness and generalization capabilities.

The qualitative analysis revealed inconsistent texture in some expanded regions, caused by the diverse sources in the training data. Incorporating domain-specific pre- and post-processing steps, such as binarization or skeletization, may reduce these artifacts. Moreover, an in-depth analysis of spurious and reconstructed minutiae would better characterize their impact on matching performance.

Beyond its primary objective of fragment expansion, the proposed method has the potential to support additional functionalities within fingerprint recognition systems. One particularly relevant application is fingerprint template construction in scenarios where multiple partial captures are combined through fragment stitching. By expanding individual fragments prior to the stitching process, the method could enhance the overlap between fragments, improve alignment accuracy, and ultimately lead to higher-quality composite templates. This, in turn, may reduce the number of captures required from the user, thereby improving usability and reducing acquisition time. Furthermore, the ability to reliably expand partial fingerprints opens the possibility of designing smaller and more cost-effective sensors, as reduced sensing areas could be compensated by computational expansion. This could have significant implications for the development of compact biometric devices, particularly in mobile and embedded systems, where space and power constraints are critical considerations.

5. Conclusion

In this work, we proposed a method for fingerprint enhancement aimed at improving partial fingerprint authentication systems by expanding fingerprint fragments using an outpainting technique with a Spectral Normalization PatchGAN model, fine-tuned for fingerprints. The proposed FETI model learns ridge and valley patterns, enabling it to expand fingerprint fragments and, in some cases, generate minutiae that correspond correctly to existing minutiae in the original fingerprint template.

Our experiments show that fragment expansion reduces the Equal Error Rate by approximately 0.524% with *Bozorth3* and 0.34% with *VeriFinger*, on average. The results suggest an optimal expansion range of 15 to 45 pixels per side, which balances improved matching performance across both Synthetic and FVC datasets. These results suggest that the positive impact of generated minutiae that correspond to existing ones may outweigh the negative impact of spurious ones in these matching tools. Expanding beyond this range may degrade performance due to spurious minutiae and texture inconsistencies, though quality scores (NFIQ 2.0) remain acceptable even at larger expansions, with an average improvement of 7.38 and 5.38 points for the synthetic and FVC datasets, respectively.

While further validation on larger datasets of real fingerprints, ablation studies, and architecture improvements would strengthen this proposal, the results demonstrate that fragment expansion is an effective approach for fingerprint enhancement. Fragment expansion represents a promising direction with measurable improvements in matching performance and quality.

Acknowledgements

The results presented in this work were funded by Samsung Eletrônica da Amazônia Ltda., under the terms of the Brazilian Informatics Law (Lei de Informática nº 8.248/91).

References

[1] Kumud Arora and Poonam Garg. A quantitative survey of various fingerprint enhancement techniques. *International Journal of Computer Applications*, 28(5):24–28, 2011. 2

[2] Kai Cao and Anil K. Jain. Latent orientation field estimation via convolutional neural network. In *2015 International Conference on Biometrics (ICB)*, pages 349–356, 2015. 2

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 3

[4] D Ezhilmaran and M Adhiyaman. A review study on fingerprint image enhancement techniques. *International Journal of Computer Science & Engineering Technology (IJCSET) ISSN*, pages 2229–3345, 2014. 2

[5] Lin Hong, Yifei Wan, and Anil Jain. Fingerprint image enhancement: algorithm and performance evaluation. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):777–789, 1998. 2

[6] Xijie Huang, Peng Qian, and Manhua Liu. Latent fingerprint image enhancement based on progressive generative adversarial network. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3481–3489, 2020. 2

[7] Anil Jain and Sharath Pankanti. Fingerprint classification and matching. *Handbook for Image and Video Processing*, pages 821–836, 2000. 2

[8] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European conference on Computer Vision (ECCV*, pages 702–716. Springer, 2016. 4

[9] Youzhi Liang and Wen Liang. Reswcae: Biometric pattern image denoising using residual wavelet-conditioned autoencoder, 2023. 2

[10] Davide Maltoni, Dario Maio, Anil K Jain, Salil Prabhakar, et al. *Handbook of fingerprint recognition*. Springer, 2009. 2

[11] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018. 4

[12] Darshan Charan Nayak. Comparative study of various enhancement techniques for finger print images. *IJCSIT) International Journal of Computer Science and Information Technologies*, 6(2):1900–1905, 2015. 2

[13] Dinh-Luan Nguyen, Kai Cao, and Anil K. Jain. Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge, 2017. 2

[14] Ramakrishna Prabhu, Xiaojing Yu, Zhangyang Wang, Ding Liu, Anxiao, and Jiang. U-finger: Multi-scale dilated convolutional network for fingerprint image denoising and inpainting, 2018. 2

[15] Aditi Roy, Nasir Memon, and Arun Ross. Masterprint: Exploring the vulnerability of partial fingerprint-based authentication systems. *IEEE Transactions on Information Forensics and Security*, 12(9):2013–2025, 2017. 1

[16] Patrick Schuch, Simon Schulz, and Christoph Busch. Survey on the impact of fingerprint image enhancement. *IET Biometrics*, 7(2):102–115, 2018. 1, 2

[17] Yao Tang, Fei Gao, Jufu Feng, and Yuhang Liu. Fingernet: An unified deep network for fingerprint minutiae extraction, 2017. 2

[18] Sukesh Adiga V and Jayanthi Sivaswamy. Fpd-m-net: Fingerprint image denoising and inpainting using m-net based convolutional neural networks, 2019. 2

[19] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018. 4

[20] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Free-form image inpainting with gated convolution. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4470–4479. IEEE, 2019. 1, 2, 3, 5

[21] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 5

[22] Yanming Zhu, Xuefei Yin, and Jiankun Hu. Fingergan: A constrained fingerprint generation scheme for latent fingerprint enhancement, 2022. 2