IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 26 September 2024, accepted 13 October 2024, date of publication 16 October 2024, date of current version 29 October 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3481507*

RESEARCH ARTICLE

# No Feature Left Behind: Filling the Gap in Fingerprint Frequency Estimation

**RAFFAELE CAPPELLI**
Department of Computer Science and Engineering, University of Bologna, 40126 Bologna, Italy
e-mail: raffaele.cappelli@unibo.it

**ABSTRACT** Fingerprint recognition remains a cornerstone of biometric identification systems. However, the critical task of ridge-line frequency estimation has been surprisingly underexplored. To address this research gap, I introduce the first publicly available benchmark for fingerprint frequency estimation, a pioneering resource that includes both high- and low-quality fingerprints with meticulously labeled ground truth features: segmentation masks, orientation fields, and frequency maps. Furthermore, two novel frequency estimation methods are proposed: one that significantly enhances a well-known frequency estimation method based on traditional image processing techniques and another that, for the first time, applies deep learning to this context. Experimental results on the new benchmark demonstrate that both methods surpass existing state-of-the-art techniques, particularly in challenging low-quality fingerprint scenarios. By providing an open-source implementation and a comprehensive benchmark, this work sets a new standard for the evaluation of frequency estimation methods, fostering further research and development in this crucial area of fingerprint recognition.

**INDEX TERMS** Benchmark testing, biometrics (access control), convolutional neural networks, deep learning, feature extraction, fingerprint recognition, frequency estimation, image processing, open source software, Python.

## I. INTRODUCTION

Fingerprint recognition is a pivotal biometric technology, integral to modern personal identification systems due to its reliability and accuracy [1], [2]. Its applications span various domains, from criminal investigations and border control to mobile device authentication, underscoring its versatility and importance.

In the realm of automated fingerprint processing, the initial steps — segmentation, orientation field estimation, and ridge-line frequency estimation (Fig. 1) — are crucial. Segmentation isolates the fingerprint pattern from the background, eliminating noise and irrelevant information, thereby enhancing the accuracy of subsequent steps. Orientation field estimation delivers essential directional information, while ridge-line frequency estimation determines local ridge/valley spacing. Both are vital for the performance of various fingerprint processing tasks, such as enhancement [1].

The local ridge-line frequency (also known as ridge-line density) at a given position is the number of ridges per unit length along a hypothetical segment orthogonal to the local ridge orientation. It can be determined by measuring the distance between two consecutive ridges or valleys (see Fig. 2). Despite its significance, ridge-line frequency estimation has garnered considerably less attention in the scientific literature compared to segmentation and orientation field estimation. While over a hundred papers have been published on each of the latter two topics [3], [4], only a dozen focus on ridge-line frequency estimation. The methods available in the scientific literature date back at least ten years. While they have achieved some success, their performance, particularly with low-quality fingerprints, remains suboptimal. This paper explores the reasons behind these shortcomings, positing that one of the primary causes is the lack of a dedicated benchmark and the apparent absence of deep-learning techniques specifically designed for this task, likely due to the unavailability of datasets with annotated frequency ground truth for training.

The associate editor coordinating the review of this manuscript and approving it for publication was Gerard-Andre Capolino.

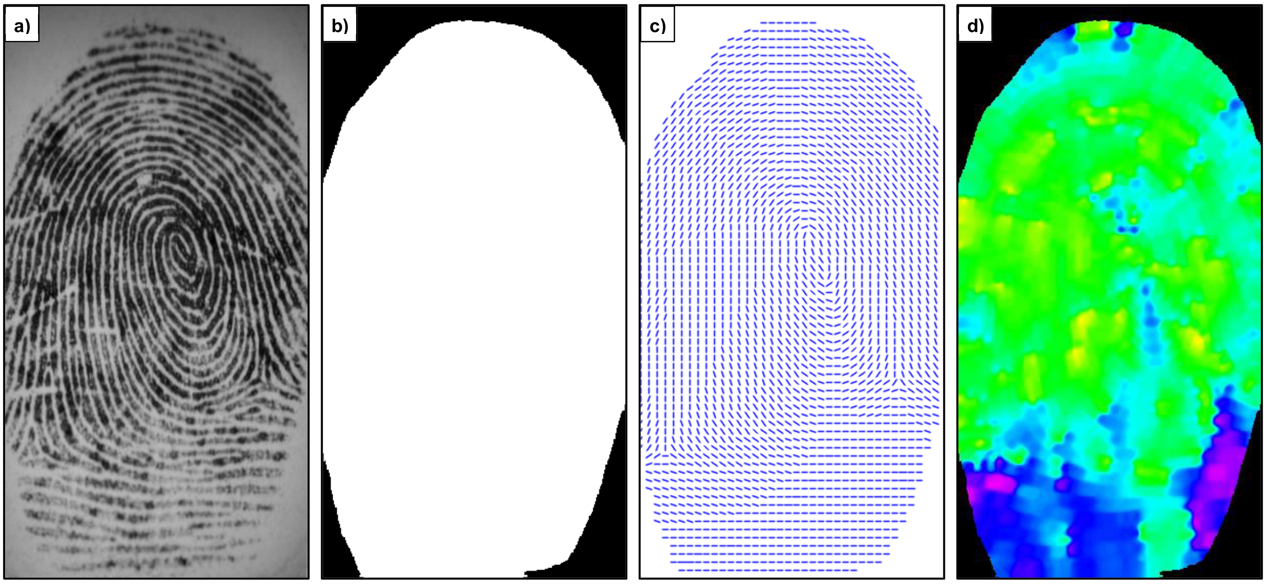


**FIGURE 1.** Example of a fingerprint with its extracted features: (a) a fingerprint, (b) its segmentation mask, (c) its orientation field (downsampled to $1/64^{th}$ of its original resolution, with line segments indicating local orientations every eight pixels), and (d) its frequency map, visualized as a color image according to the color scale in Fig. 2.

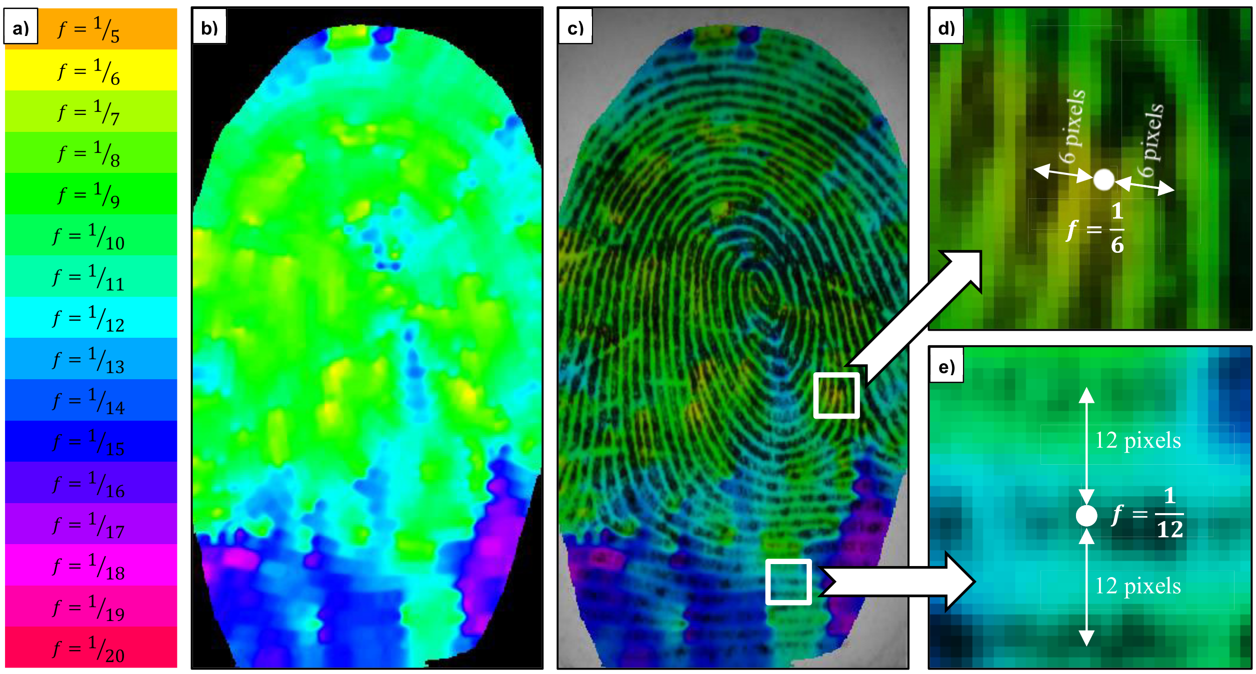


**FIGURE 2.** Frequency visualization and interpretation: (a) color map used to represent frequency values, (b) frequency image, (c) frequency image superimposed on the corresponding fingerprint, and (d)-(e) magnified fingerprint regions illustrating the relationship between frequency and ridge-line spacing.

This paper addresses this gap with the following three main contributions:

- A benchmark for Fingerprint Frequency Estimation (FFE), featuring both high- and low-quality fingerprints with manually labeled ground truth features, including segmentation masks, orientation fields, and frequency maps.
- Two frequency estimation methods that surpass current state-of-the-art approaches, one leveraging straightforward image processing techniques and the other utilizing deep learning.
- An open-source implementation of both methods.

The significance of these contributions lies in their novelty and their potential to standardize the evaluation of frequency estimation methods, thereby stimulating further research in this underexplored area, as described below.

- The FFE benchmark is the first publicly available benchmark for evaluating fingerprint frequency estimation methods. Thanks to careful manual markup of features and the definition of an appropriate evaluation metric, it allows for the first time a direct and unbiased assessment of fingerprint frequency estimation techniques.
- The two proposed frequency estimation methods not only serve as examples of using the FFE benchmark but also constitute original research contributions. The first method revisits a traditional technique, adding substantial modifications that greatly improve its accuracy. The second method is the first published work to utilize deep learning for the precise estimation of local fingerprint frequencies.
- By providing a comprehensive benchmark with publicly available images and annotated features, and by offering an open-source implementation of the two novel methods, this work can effectively bridge the gap in the literature and encourage the development of more robust and accurate frequency estimation techniques.

Both the benchmark and the novel methods adhere to the KISS (Keep It Simple and Straightforward[1]) principle [5], [6]: they have been designed to be as simple and accessible as possible to encourage widespread adoption.

The rest of this paper is organized as follows: Section II reviews the main fingerprint frequency estimation methods, highlighting the disparity in research focus and the lack of recent advancements. Section III introduces the new FFE benchmark. Sections IV and V describe the two novel frequency estimation methods, respectively. Section VI reports experiments aimed at evaluating the performance of the proposed methods and comparing them to the state-of-the-art on the FFE benchmark. Finally, Section VII draws some concluding remarks.

## II. RELATED WORKS

Fingerprint frequency estimation algorithms can be broadly classified into two categories: *spatial-domain methods* and *Fourier-domain methods*.

### A. SPATIAL-DOMAIN METHODS

Spatial-domain methods aim to directly measure ridge-line spacing from the input image, which is typically divided into equally-sized blocks. From each block, an estimated local frequency value is obtained. The approach proposed in [7] is based on the *x-signature* of the ridges and valleys within a 32 × 16 window, rotated according to the local ridge-line orientation (see Fig. 3). This simple method is effective in high- and medium-quality fingerprint regions, but often fails to reliably estimate the correct frequency in low-quality regions. The authors use interpolation and low-pass filtering to deal with such cases. Other x-signature-based methods are introduced in [8], [9], [10], and [11]. In [8], the author adopts a specific form of higher-order spectrum to enhance the fundamental frequency and suppress noise. The method described in [9] fits a Chebyshev polynomial to the x-signature, then uses the first and second derivatives to estimate ridge and valley width. The author of [11] argues that most frequency estimation errors occur in highly-curved regions, where a single dominant orientation cannot be defined. Therefore, he proposes to locally rectify ridges based on ridge curvature before computing the x-signature. Other spatial-domain methods include [12], [13], and [14]. In [12], the ridge pattern is locally modeled as a sinusoidal-shaped surface, and the variation theorem is exploited to estimate the unknown frequency. Reference [13] adopts scale-space theory to locally estimate the ridge width, relying on combinations of normalized derivatives computed pointwise. Reference [14] describes a geometrical method where gray-level minima (assumed to lie on ridges) are used as vertices for a Delaunay triangulation, whose resulting graph is parsed to estimate the distances among adjacent ridges.

### B. FOURIER-DOMAIN METHODS

Fourier-domain methods typically divide the image into square blocks (e.g., 32 × 32) and apply a Discrete Fourier Transform (DFT) to each block. If the ridge-line orientations and frequencies are uniform within the block, the pattern can be modeled as a surface wave, and its Fourier spectrum exhibits two symmetric peaks, whose distance from the origin indicates the frequency (see Fig. 4). Reference [15] introduces a Fourier-domain method to simultaneously compute local orientation and frequency from each block; the estimation is probabilistic and thus quite robust to outliers. Reference [16] focuses solely on frequency estimation and proposes to directly search for the maxima in the Fourier spectrum of each block.

Two additional papers compare methods from the two categories. Reference [17] reports no significant differences between a spatial-domain method based on geometric measurements on the image and a Fourier-domain method based on a search of the maxima in the spectrum. Conversely, [18] reports that Fourier-domain methods seem to be more robust against noise.

### C. FILLING THE GAP

Analyzing the scientific literature reveals a striking disparity between the number of methods proposed for frequency estimation (approximately a dozen) and those proposed for background segmentation and orientation estimation, which exceed 100 for each category [3], [4]. This imbalance is noteworthy and suggests that frequency estimation has not received the same level of attention.

[1] While other interpretations of the KISS acronym are more common, this paper employs a less colloquial but more academically appropriate definition in the interest of maintaining scientific precision and avoiding potential misinterpretations.

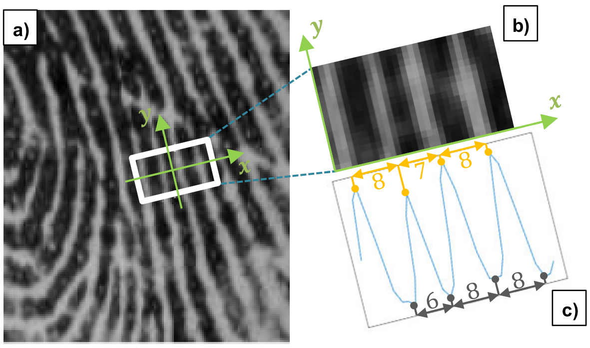


**FIGURE 3.** Example of x-signature computation. (a) A fingerprint. (b) A 32 × 16 window with the $y$ axis aligned to the local orientation. (c) The x-signature (plotted in blue) is the sequence of values obtained by averaging the window along the $y$ axis to reduce noise; its maxima and minima correspond to the valleys and ridges, respectively. The local frequency can be estimated as the inverse of the average distance between consecutive maxima ($f = \frac{3}{8+7+8} \cong \frac{1}{7.7}$) or minima ($f = \frac{3}{6+8+8} \cong \frac{1}{7.3}$).

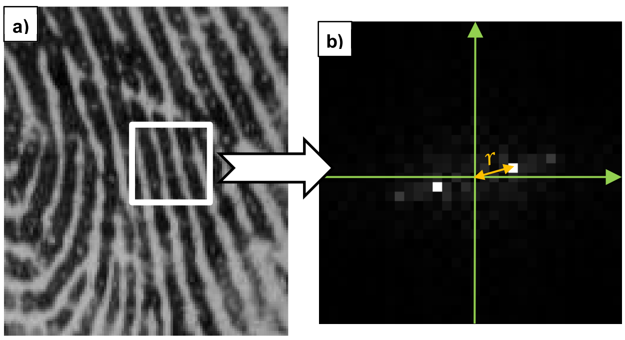


**FIGURE 4.** Example of local frequency estimation in the Fourier domain. (a) A fingerprint with a 32 × 32 window highlighted. (b) The DFT is computed over the window. The distance $r$ between one of the two peaks and the center of the spectrum can be used to estimate the local frequency $f = \frac{r}{32} = \frac{\sqrt{16+1}}{32} \cong \frac{1}{7.8}$.

Examining the dates of the respective studies cited in the previous paragraphs, it is evident that no work has been published after 2014. To the author's knowledge, no new methods for fingerprint frequency estimation have been published in IEEE journals in the last ten years. Yet, the problem of frequency estimation does not appear to be significantly simpler than orientation estimation, nor can it be considered fully solved. In low-quality fingerprint images, none of the existing methods seem to produce satisfactory results. This indicates that frequency estimation remains a challenging task and deserves further research.

For the other two problems, public benchmarks are available (see [19] for segmentation, and [20], [21] and [22] for orientation estimation), which recent papers use for comparison. However, no such benchmark exists for frequency estimation. The evaluation of results in related papers is highly heterogeneous: some methods consider the biometric verification performance of a complete fingerprint recognition system, others use synthetic images with known frequencies, some provide only qualitative assessments on a few images, and others use manually labeled frequencies by experts without making them publicly available.

It is reasonable to hypothesize that the lack of a public and specific benchmark for frequency estimation (i.e., a dataset of fingerprints with manually labeled ground-truth frequencies) is one of the main factors limiting research in this area. Additionally, the absence of new studies after 2014, a period shortly following the renewed interest in neural networks, does not seem coincidental. Neural networks have been successfully applied in various aspects of fingerprint recognition over the past decade [23], [24], [25], but not for local frequency estimation, despite the unsatisfactory performance of existing methods. Given that neural networks require labeled ground-truth data for training and testing, it is believed that the lack of such data and specific benchmarks for local frequency estimation is the primary reason for the absence of studies in the past ten years. However, creating such a benchmark is not straightforward: while local orientations can be visually represented as segments overlaid on the fingerprint, making it possible to develop software tools for manual markup, frequency estimation is much more complex to visualize and annotate.

These observations led to the conception of this work:
- To develop a dedicated benchmark for evaluating and comparing frequency estimation methods, which will be made publicly available.
- To design two new methods and compare them, on the new benchmark, with state-of-the-art spatial-domain and Fourier-domain methods.

## III. THE FFE BENCHMARK

The FFE benchmark is the first dedicated resource for evaluating frequency estimation methods. It contains 60 fingerprints acquired with optical scanners at 500 dpi resolution, with image sizes ranging from 328 × 364 pixels to 448 × 560 pixels. The fingerprints are divided into a "Good" dataset with 10 high-quality fingerprints and a "Bad" dataset with 50 low-quality fingerprints, which include those with cuts, abrasions, and dry or wet skin conditions. For each fingerprint, the benchmark provides the following ground truth data: segmentation mask, orientation field, and frequency map (see Figures 5 and 6). By including segmentation and orientation ground truth, the benchmark allows for the assessment of the performance of frequency estimation methods independently of potential biases introduced by other processing steps.

Drawing inspiration from the KISS principle, the FFE benchmark has been designed with the following characteristics:
- *Dense pixel-wise ground truth* – unlike most annotated fingerprint datasets, features are not provided on a sparse grid. Every pixel has its corresponding ground truth data, offering richer information for evaluation.
- *Simple and familiar storage format* – all data is stored in the widely used Portable Network Graphics (PNG)

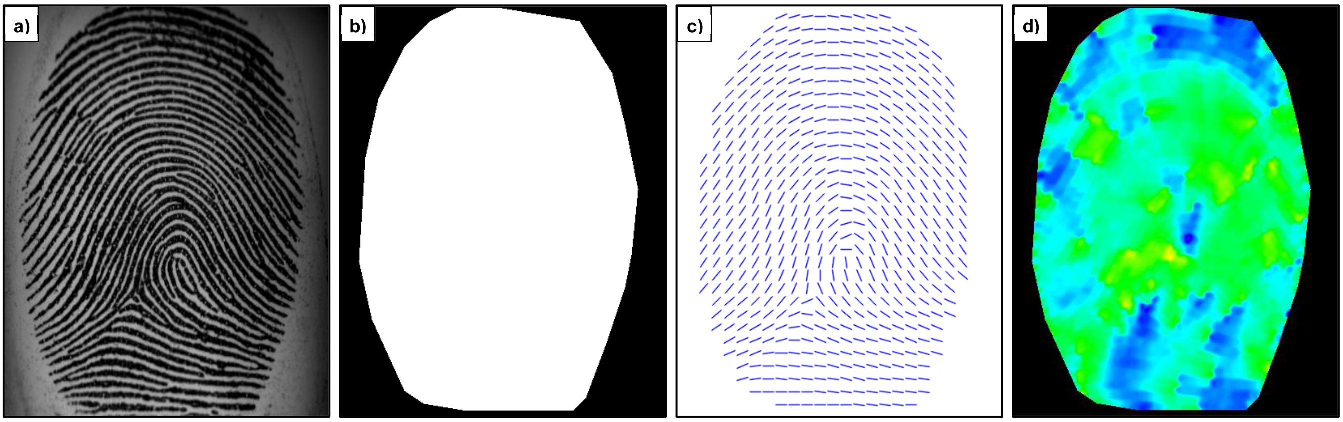


**FIGURE 5.** Sample from the FFE "Good" dataset: (a) fingerprint image, (b) ground-truth segmentation mask, (c) ground-truth orientation field (displayed with line segments indicating local orientations every 16 pixels), and (d) ground-truth frequency map, visualized according to the color scale in Fig. 2.

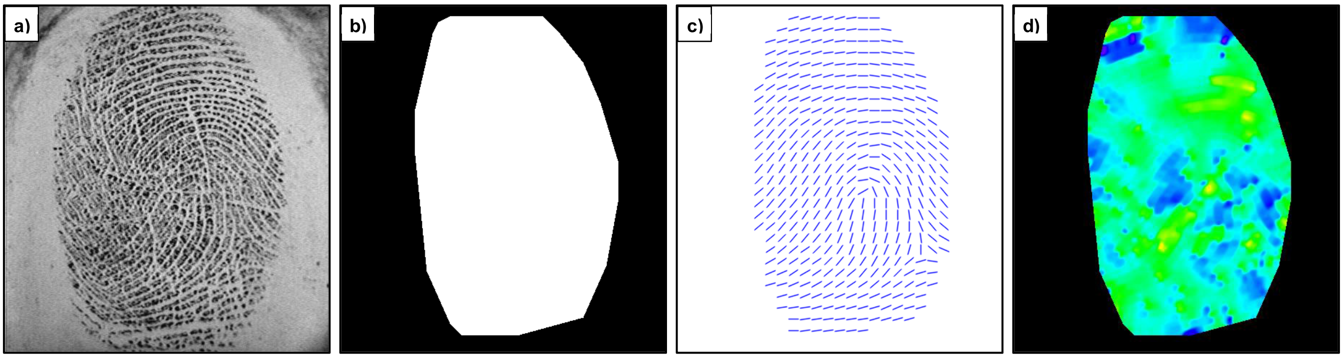


**FIGURE 6.** Sample from the FFE "Bad" dataset: (a) fingerprint image, (b) ground-truth segmentation mask, (c) ground-truth orientation field (displayed with line segments indicating local orientations every 16 pixels), and (d) ground-truth frequency map, visualized according to the color scale in Fig. 2.

format. In particular, for each fingerprint, the benchmark contains four grayscale images with one byte per pixel. This simplifies access and integration with existing tools and software libraries.
- *Straightforward evaluation metric*– The benchmark utilizes the Mean Absolute Percentage Error (MAPE) for easy-to-understand and readily implementable performance assessment.

### A. EVALUATION METRIC

The FFE benchmark leverages the Mean Absolute Percentage Error (MAPE) to quantify the accuracy of frequency estimation methods. This choice is motivated by the practical significance of frequency estimation errors in subsequent fingerprint processing steps. An error of one pixel on a ridge-line spacing of five pixels is significantly more impactful than on a ridge-line spacing of ten pixels.

$$MAPE\left(\mathcal{F}, \widehat{\mathcal{F}}, \mathbf{S}\right) = \frac{\sum_{i,j} \left| \frac{1}{\mathcal{F}_{i,j}} - \frac{1}{\widehat{\mathcal{F}}_{i,j}} \right| \cdot \widehat{\mathcal{F}}_{i,j} \cdot \mathbf{S}_{i,j}}{\sum_{i,j} \mathbf{S}_{i,j}} \tag{1}$$

where $\mathcal{F}$ represents the estimated frequency map, $\widehat{\mathcal{F}}$ denotes the ground truth frequency map, and $\mathbf{S}$ is the ground truth segmentation mask ($\mathbf{S}_{i,j}$ is one for foreground pixels, zero otherwise). Notably, errors are computed on the inverse of frequencies, corresponding to ridge-line spacing.

The accuracy for each dataset ("Good" or "Bad") is reported as the average MAPE, expressed as a percentage. The straightforward MAPE metric is particularly appropriate in this context due to the availability of manually annotated ground truth data. In the absence of such ground truth, other studies have resorted to metrics such as signal-to-noise ratio (SNR), as seen in [25] where SNR is used to measure the local ridge-line quality of latent fingerprints.

### B. STORAGE FORMAT

For each fingerprint, four grayscale PNG images with one byte per pixel, are available:

- The fingerprint image itself.
- The ground truth segmentation mask, where pixel values are either zero (background) or 255 (foreground).
- The ground truth orientation field, where each pixel value $v \in [0, 255]$ corresponds to an angle $\theta = \frac{\pi \cdot v}{255}$ radians.
- The ground truth frequency map, where each pixel value $v \in [1, 255]$ corresponds to a frequency $f = \frac{10}{v}$.

### C. SOURCE DATA AND MARKUP WORKFLOW

The source data of the FFE benchmark originates from the publicly available FOE-TEST benchmark [21]. While FFE utilizes the same set of fingerprints (10 fingerprints in the "Good" dataset and 50 in the "Bad" dataset), the ground truth data for segmentation maps and orientation fields differs. FOE-TEST provides this data on a sparse grid (every 8 pixels), whereas FFE offers dense, pixel-wise information. To achieve this, the segmentation masks and orientation fields were upsampled using interpolation techniques, followed by meticulous manual verification and correction facilitated by custom software tools.

Generating ground truth for frequency maps presents a unique challenge. While it is relatively straightforward to develop software tools that display segments corresponding to orientations overlaid on the fingerprint and allow users to modify them, no suitable graphical representation has been found for intuitive frequency editing. Therefore, another feature was utilized: the ridge-line skeleton [1]. The skeleton can be easily visualized on the fingerprint (Fig. 7) and was manually edited, for each fingerprint, with a custom-developed software tool. Once the skeleton ground truth had been manually marked up, the frequency was automatically determined, for almost all the pixels of the skeleton, by measuring the distance to the two closest ridges along the direction orthogonal to the local orientation. Subsequently, an image inpainting technique [26] was employed to derive local frequency values for all the other pixels (see Fig. 7). Rigorous manual checks confirmed the accuracy of this approach.

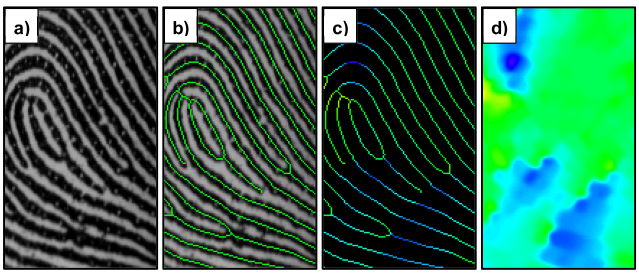


**FIGURE 7.** Process of obtaining the frequency ground truth: (a) portion of a fingerprint; (b) manual markup of its ridge-line skeleton; (c) frequency estimated on the pixels of the skeleton; (d) final frequency ground truth, visualized according to the color scale in Fig. 2.

## IV. THE XSFFE METHOD

XSFFE (X-Signature-based Fingerprint Frequency Estimation) is a spatial-domain method based on traditional image processing techniques and on the computation of the x-signature. XSFFE adheres to the KISS principle: it consists of a simple sequence of well-known image processing operations, readily available in computer vision libraries such as OpenCV [27], and can be implemented in less than 50 lines of straightforward Python code. Its processing steps (Fig. 8) are described below.

- *Preprocessing* – This step applies median filtering and Gaussian smoothing, which reduce noise and help to produce smoother x-signatures.

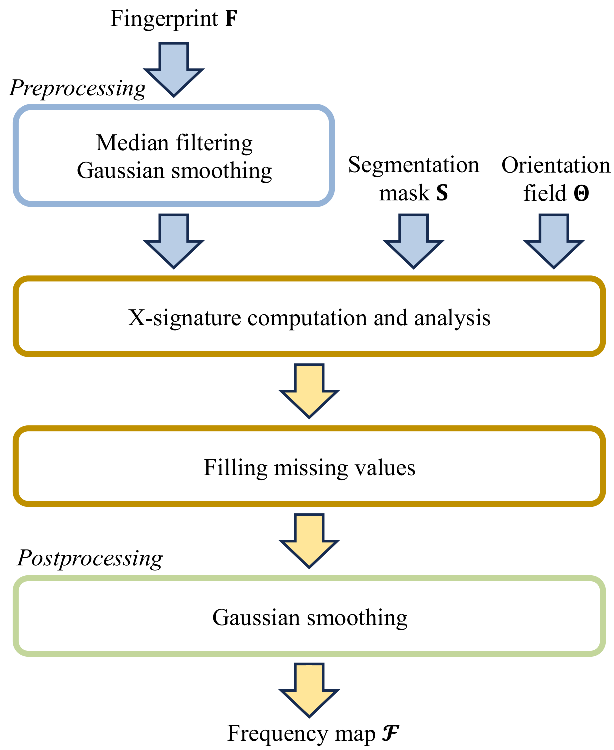


**FIGURE 8.** Overview of the XSFFE method. This figure illustrates the input data (fingerprint, segmentation mask, and orientation field), the sequential processing steps, and the resulting frequency map.

- *X-signature computation* – The x-signature is computed on a $23 \times 43$ window, only on foreground pixels that are at least 11 pixels away from the background, as regions near the background are often of low quality and may lead to incorrect estimates. These two parameters are crucial because they balance estimation accuracy and noise robustness. A window that is too small will be affected by local noise, such as cuts or abrasions on the fingertip, whereas a window that is too large will reduce accuracy in high-curvature areas. Similarly, the distance from the background helps ignore regions that often contain much of the noise, but a value that is too large will reduce overall accuracy. Initial tests on a separate dataset of fingerprints helped determine the optimal values for these parameters (more details at the end of this section).
- *X-signature analysis* – For each pixel analyzed, both minima and maxima of its x-signature are examined: the distances between consecutive maxima denote the spacing between valleys, while the distances between consecutive minima denote the spacing between ridges. By considering both types of distances, the method achieves a more robust estimate. To discard outliers, the median distance is computed first, and any distance differing by more than two pixels from the median is identified as an outlier and discarded. If at least four distances remain, the frequency is estimated as the inverse of their average.

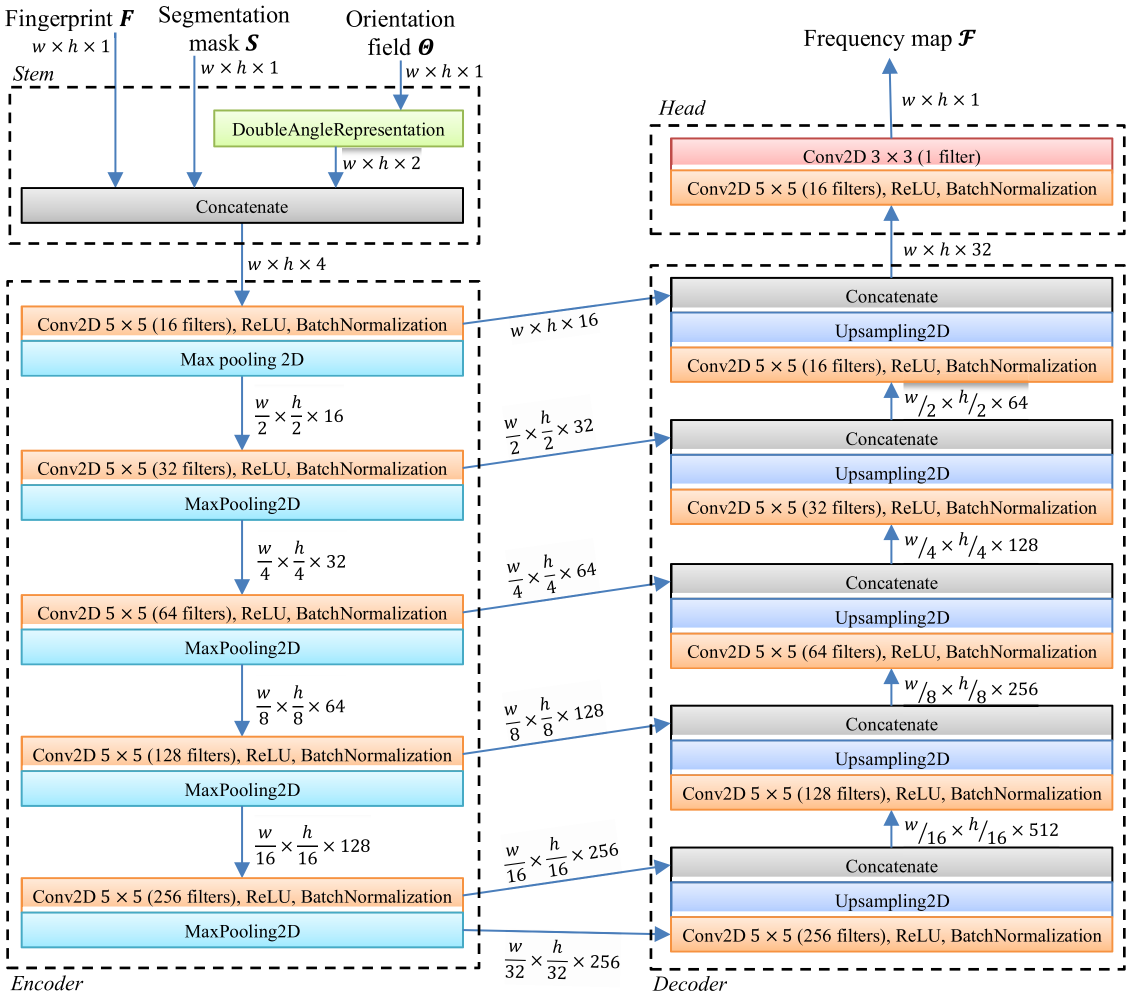


**FIGURE 9.** **Overview of the SNFFE network architecture. This figure illustrates the input data (fingerprint, segmentation mask, and orientation field), the main building blocks (Stem, Encoder, Decoder, Head), and the resulting frequency map.**

- *Filling missing values* – Any value that could not be computed in the previous step, either because too close to the background or because not enough valid distances were found, is determined using the inpainting technique proposed in [26].
- *Postprocessing* – A final Gaussian filter is applied to smooth the frequency map.

To reduce computation time, the x-signature is calculated every eight pixels, then the frequency map is upsampled to the image resolution. This approach allows for a 64× speed-up without significantly reducing the accuracy, due to the smooth nature of the frequency map. The values for the various parameters of this method have been chosen during some preliminary experiments on fingerprints from FVC2002 [28] and FVC2004 [29] databases (see table 1), with frequency ground truth manually obtained following the same workflow described in section III-C. Results of this method on the FFE benchmark are reported in section VI.

## V. THE SNFFE METHOD

SNFFE (Simple Network for Fingerprint Frequency Estimation) is, to the best of the author's knowledge, the first fingerprint frequency estimation method based on deep learning.

The architecture of the SNFFE network (Fig. 9) is similar to that of the network proposed in [4] for fingerprint orientation field estimation. That network model was meticulously chosen in [4] as the most promising design after evaluating several alternatives inspired by successful convolutional models like ResNet [30], U-Net [31], DeepLabv3+ [32], and ConvNext [33]. Since orientation and frequency estimation are strictly related, it is reasonable to assume that an architecture so successful in orientation field estimation can achieve interesting results in frequency estimation as well. The validity of this choice was confirmed by some preliminary experiments on a separate test set (see Table 1) and by the final evaluation on the FFE benchmark (see section VI).

**TABLE 1.** Fingerprint datasets used for parameter tuning, training, and evaluation.

| Dataset | Fingerprints | Usage for XSFFE | Usage for SNFFE |
|---|---|---|---|
| Subset of FVC2002 DB2 A (third impression of each finger) | 100 | Parameter tuning | Training |
| Subset of FVC2002 DB3 A (third impression of each finger) | 100 | Parameter tuning | Training |
| Subset of FVC2004 DB1 A (first impression of each finger) | 100 | Parameter tuning | Training |
| Subset of FVC2004 DB3 A (third impression of each finger) | 100 | - | Network model evaluation |
| FFE benchmark ("Good" and "Bad" datasets) | 60 | Evaluation | Evaluation |

**FIGURE 10.** Examples of the data augmentation techniques applied to the training fingerprints.

The network comprises four key building blocks: the Stem, the Encoder, the Decoder, and the Head (Fig. 9). SNFFE takes a three-channel image as input, containing the fingerprint $\mathbf{F}$, its segmentation mask $\mathbf{S}$, and its orientation field $\mathbf{\Theta}$. The Stem block converts the orientation field $\mathbf{\Theta}$, containing an angle in radians $\mathbf{\Theta}_{i,j} \in [0, \pi]$ for each pixel $(i, j)$, into a double-angle representation:

$$\mathbf{X}_{i,j} = \cos\left(2 \cdot \mathbf{\Theta}_{i,j}\right), \mathbf{Y}_{i,j} = \sin\left(2 \cdot \mathbf{\Theta}_{i,j}\right) \tag{2}$$

This representation of the input orientations, according to some preliminary experiments, improves the convergence and accuracy of the network.

Similarly to the network introduced in [4], both the Encoder and Decoder are symmetrical, each with five levels. Each encoder level employs a $5 \times 5$ convolution with padding and ReLU activation, followed by batch normalization and $2 \times 2$ max pooling for downsampling. Conversely, each decoder level utilizes a $5 \times 5$ convolution with padding and ReLU activation, batch normalization, and a $2 \times 2$ spatial upsampling layer. Skip connections play a crucial role in information flow. They deliver feature maps obtained after the batch normalization layer of each encoder level to the corresponding decoder level, where they are concatenated with the upsampled features.

The Head block operates at the same resolution of the input fingerprint and applies 16 $5\times5$ convolution filters with padding, ReLU activation, and batch normalization, generating 16 feature maps. A final $3\times3$ convolution layer with linear activation yields the frequency values. In fact, the output of the network is an image containing the inverse of the frequency for each pixel.

The loss function used to train the network is the MAPE function (1). To train the network, 300 fingerprints were sourced from the FVC2002 [28] and FVC2004 [29] databases (Table 1). The frequency ground truth was manually obtained following the workflow described in Section III-C.

Preliminary experiments were conducted using an additional set of FVC fingerprints (Table 1) as a test set. It is important to note that using the FFE benchmark dataset for model and hyperparameter selection would introduce bias, as the model would be indirectly optimized for that specific test set rather than for generalization to new data. Therefore, the FFE benchmark is used only once to evaluate the final performance of SNFFE (see Section VI). These preliminary experiments confirmed that the training procedure and hyperparameters adopted in [4] were generally appropriate for this problem as well, with two exceptions: the number of fingerprints in each batch and the number of batches per epoch. In fact, better results were obtained by increasing these two hyperparameters from 16 to 32 and from 120 to 300, respectively. A possible explanation for why it was beneficial to increase these two hyperparameters is that the fingerprint frequency is a smoother and less variable feature than the fingerprint orientation field. In this case, capturing global patterns becomes more important. Larger batch sizes help in aggregating information over a wider range of samples, aiding in the learning of these global patterns. Another explanation is that smoother features like the frequency might lead to overfitting if the model is trained on small batches with high variability. Increasing the batch size and the number of batches per epoch helps mitigate this risk by providing a more stable learning process.

Below is an overview of the training procedure and hyperparameters. Interested readers can find a more detailed description in [4].

- *Input image size* – all training fingerprints were padded to a uniform size of $512 \times 512$ pixels.

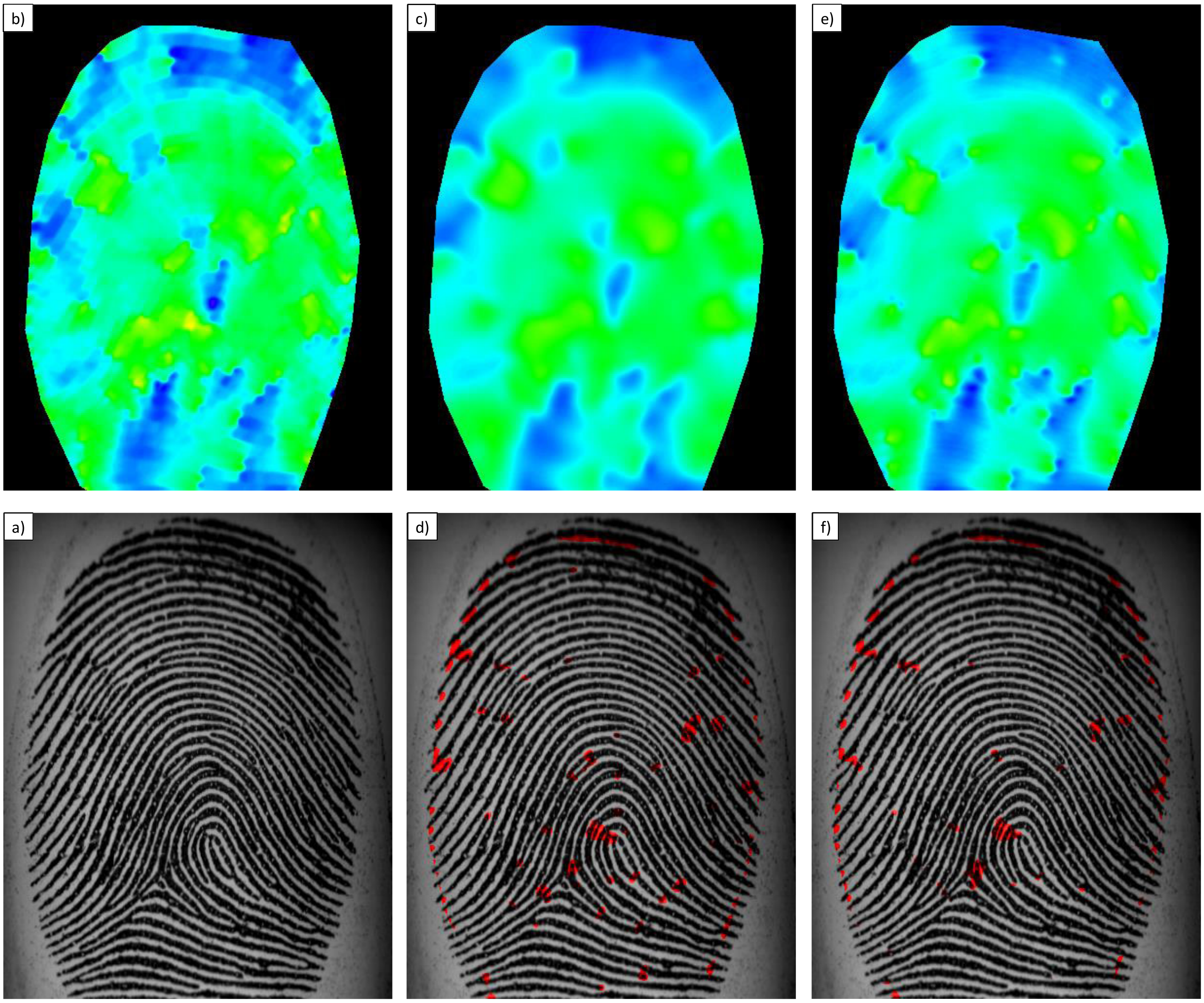


**FIGURE 11.** Example of frequency estimation on a fingerprint from the FFE "Good" dataset. (a) Fingerprint image; (b) Frequency ground truth; (c) Frequency map estimated by XSFFE (*MAPE* = 4.11%); (d) XSFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red); (e) Frequency map estimated by SNFFE (*MAPE* = 3.16%); (f) SNFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red). Frequencies are visualized according to the color scale in Fig. 2.

- *Data augmentation* – various augmentation techniques were applied, including random translation, rotation, scale, horizontal flip, gamma correction, contrast reduction, morphology operations, and simulation of scratches and abrasions (see Fig. 10).
- *Optimization algorithm* – Adam [34] with Nesterov momentum [35] was employed. The learning rate was adjusted between $10^{-5}$ and 0.025 using a cosine decay strategy with warmup. The momentum parameters were set to $\beta_1 = 0.2$ and $\beta_2 = 0.5$.
- *Training epochs* – 25 training epochs, each consisting of 300 batches of 32 fingerprints.

SNFFE adheres to the KISS principle by providing the following characteristics:

- *End-to-end frequency estimation* – the entire frequency estimation task is carried out within a single network, eliminating the need for separate preprocessing or postprocessing steps.
- *Purely convolutional architecture* – the network exclusively employs convolutional layers, allowing it to directly process fingerprints of any size[2] without requiring image resizing or mosaicking.
- *Dense pixel-wise estimation* – the output of the network is a dense frequency map, providing an estimate of the local frequency at each pixel within the fingerprint.
- *Compact network architecture* – the network is relatively compact (less than five million parameters), minimizing hardware requirements and computational cost.

[2]There is a minor constraint on the image size: both the width and height of the image must be multiples of 32. If the dimensions do not already comply, the image needs to be padded to meet this requirement.

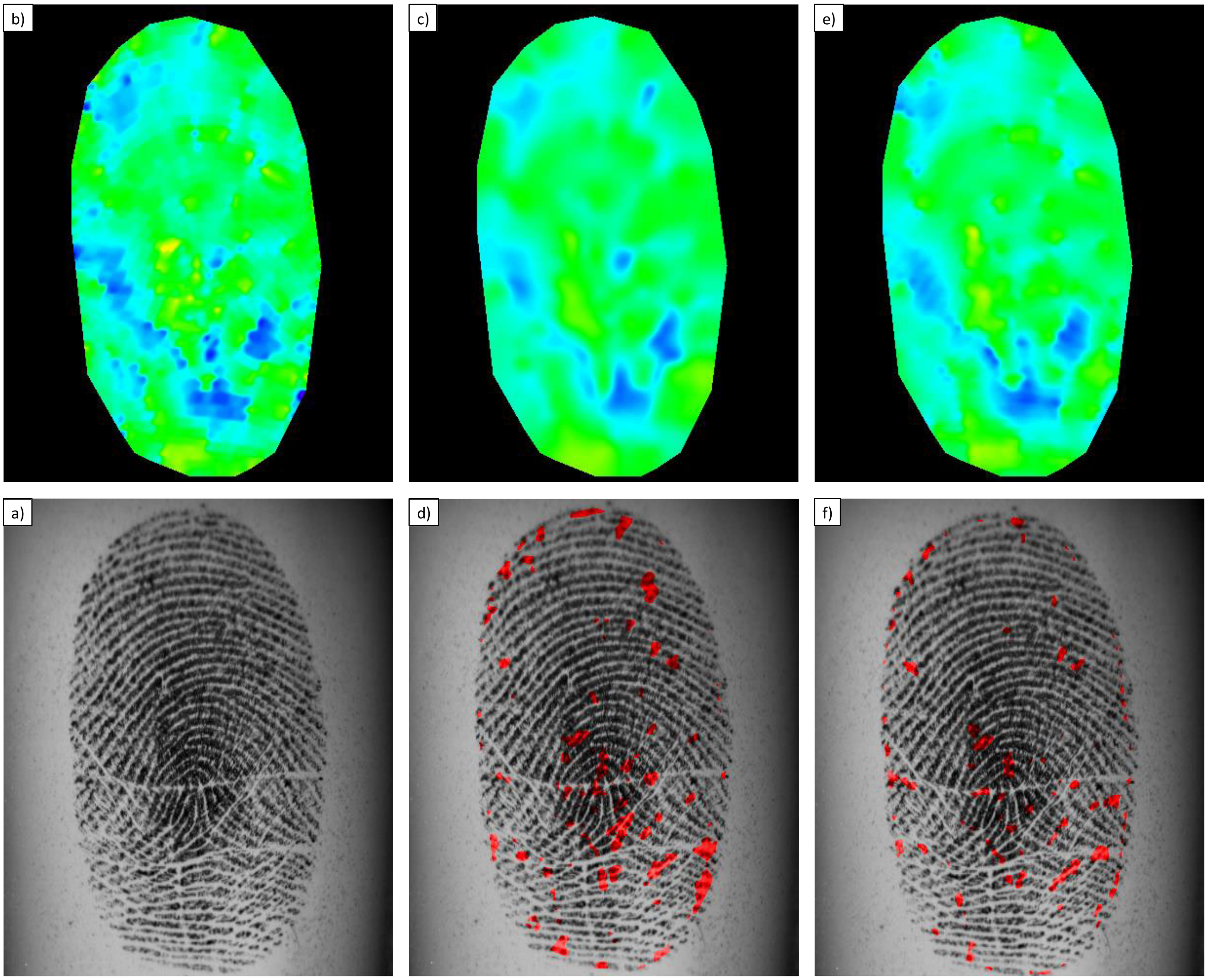


**FIGURE 12.** Example of frequency estimation on a fingerprint from the FFE "Bad" dataset. (a) Fingerprint image; (b) Frequency ground truth; (c) Frequency map estimated by XSFFE (*MAPE* = 5.15%); (d) XSFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red); (e) Frequency map estimated by SNFFE (*MAPE* = 3.97%); (f) SNFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red). Frequencies are visualized according to the color scale in Fig. 2.

- *Framework-agnostic implementation* – the network architecture employs standard layers and leverages a common optimization technique with a straightforward loss function. This facilitates easy implementation across various deep-learning frameworks.

## VI. EXPERIMENTAL RESULTS

XSFFE and SNFFE are implemented in Python using the OpenCV library [27], with SNFFE further leveraging the Keras library [36] as its deep-learning framework. The experiments have been performed on a PC with an Intel® Xeon® Silver 4112 CPU at 2.60GHz, equipped with an NVIDIA GeForce RTX™ 3080 Ti GPU.

Table 2 compares XSFFE and SNFFE to two state-of-the-art methods on FFE:

**TABLE 2.** Average MAPE on FFE (Percentage Values)†.

| Method | Dense estimation | | Estimation every 12 pixels | |
|---|---|---|---|---|
| | Good | Bad | Good | Bad |
| HONG | 8.10 | 13.14 | 8.13 | 13.17 |
| STFT | - | - | 9.09 | 11.74 |
| STFT(*) | - | - | 8.86 | 10.91 |
| XSFFE | 5.10 | 7.84 | 5.18 | 7.87 |
| SNFFE | **3.62** | **6.69** | **3.72** | **6.73** |

†Best values are in bold

- HONG – a spatial-domain algorithm proposed in [7], based on x-signature;
- STFT – a Fourier-domain algorithm proposed in [15].

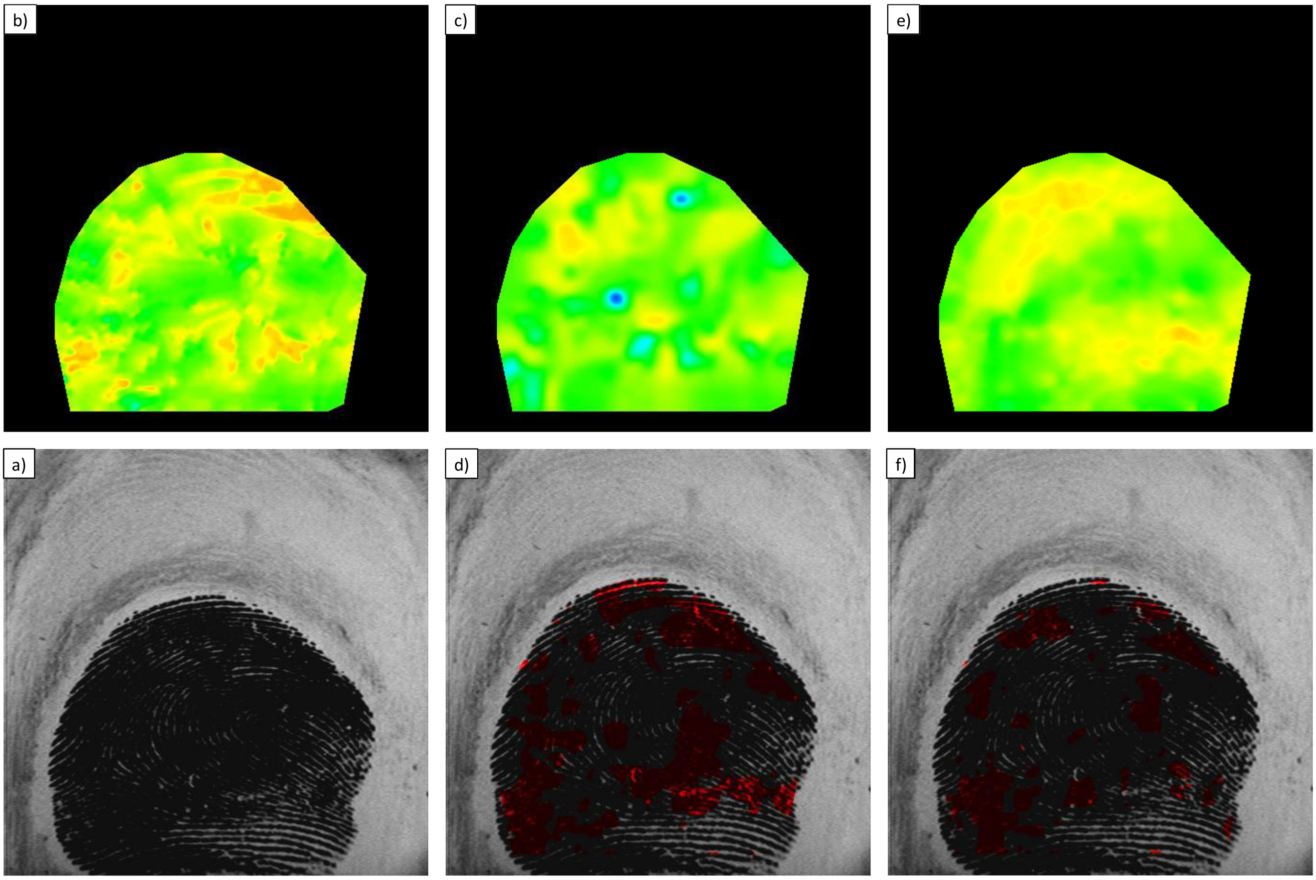


**FIGURE 13.** **Example of frequency estimation on a fingerprint from the FFE "Bad" dataset. (a) Fingerprint image; (b) Frequency ground truth; (c) Frequency map estimated by XSFFE (*MAPE* = 14.80%); (d) XSFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red); (e) Frequency map estimated by SNFFE (*MAPE* = 8.94%); (f) SNFFE errors (pixels with Absolute Percentage Error ≥ 15% are highlighted in red). Frequencies are visualized according to the color scale in Fig. 2.**

Please note that the STFT was designed to estimate frequencies on a grid (every 12 pixels). Modifying it to estimate frequencies on all pixels would significantly alter its nature. Therefore, results for other methods are reported both in the "dense" case (i.e., estimation on all pixels) and on the same grid used by STFT.

Furthermore, since STFT estimates frequencies simultaneously with the orientation field, it is not possible to simply use the FFE ground truth orientations as with other methods. For this reason, the table reports both the results of the original formulation of STFT (where the ground truth orientations are not used) and those of STFT(*), a slightly modified version that, before estimating the frequency, applies a directional filter, corresponding to the FFE ground truth orientation, to the Fourier spectrum.

Some considerations based on the results reported in Table 2:

- Of the two state-of-the-art methods, the Fourier-domain method is more accurate on low-quality fingerprints, as previously observed in [18].
- The modification made to STFT to utilize ground truth orientations is effective.
- The two new methods proposed in this work achieve significantly better results compared to the state-of-the-art methods, on both the "Good" and "Bad" datasets.
- The learning-based method SNFFE achieves the best results, although they are not vastly superior to the more traditional XSFFE method.

Fig. 11, 12, and 13 present examples of frequency estimation on FFE fingerprints. Fig. 11 shows a high-quality fingerprint from the "Good" FFE dataset. Both XSFFE and SNFFE provide accurate estimations, with a MAPE of 4.11% and 3.16%, respectively. The fingerprint in Fig. 12 is of lower quality, but both methods handle it gracefully, although XSFFE is less precise (its MAPE is 5.15%, while SNFFE achieves 3.97%). Finally, the fingerprint in Fig. 13 is more challenging: XSFFE struggles to estimate the correct frequencies in many regions, resulting in a MAPE of 14.80%, while SNFFE obtains 8.94%.

Finally, table 3 reports the average execution times for a single FFE fingerprint of the two proposed methods and the state-of-the-art approaches considered. When analyzing the execution times, it is necessary to consider that SNFFE is mostly executed on a GPU, while the others run on a CPU. Additionally, STFT and STFT(*) are MATLAB programs,

**TABLE 3.** Average execution time on FFE.

| Method | Average execution time (seconds) |
|---|---|
| HONG | 0.665 |
| STFT | 0.224 |
| STFT(*) | 0.345 |
| XSFFE | 0.308 |
| SNFFE | 0.084 |

whereas the others are written in Python. Nevertheless, it can be observed that SNFFE is significantly more efficient, while XSFFE, being entirely written in Python, including its most demanding part, which is the calculation of the x-signature at each position, is not particularly fast compared to the others.

## VII. CONCLUSION

This paper addresses a significant gap in fingerprint frequency estimation research by introducing the first publicly available benchmark for this task, the FFE benchmark. This benchmark includes both high- and low-quality fingerprints with manually labeled ground truth features, providing a comprehensive framework for the unbiased evaluation and comparison of frequency estimation methods.

Two novel frequency estimation methods are also proposed: XSFFE, which leverages straightforward image processing techniques, and SNFFE, which utilizes deep learning. Experimental results demonstrate that XSFFE and SNFFE outperform existing state-of-the-art methods, both in high-quality and low-quality fingerprints.

While the FFE benchmark and the proposed methods represent significant advancements, there are limitations to consider. The benchmark focuses on fingerprint images, and it remains to be seen how well these methods generalize to other biometric modalities with similar ridge-line patterns, such as handprints and footprints. Future research could explore the applicability of these frequency estimation techniques to these other biometric data, potentially adapting the benchmark to include a wider range of modalities.

Additionally, the proposed methods, while effective, may benefit from further refinement. For instance, integrating more sophisticated machine learning models or hybrid approaches that combine traditional image processing with deep learning could yield even better results. The introduction of the FFE benchmark[3] and the open-source[4] implementation of these methods aims to stimulate further research and development in the field of fingerprint frequency estimation. By providing a standardized evaluation framework, the goal is to encourage the development of more robust and accurate frequency estimation techniques, enhancing the overall performance of automated fingerprint recognition systems.

[3] https://github.com/raffaele-cappelli/FFE

[4] https://github.com/raffaele-cappelli/pyfing

## REFERENCES


[1] D. Maltoni, D. Maio, A. K. Jain, and J. Feng, *Handbook of Fingerprint Recognition*. Cham, Switzerland: Springer, 2022.

[2] D. Maltoni and R. Cappelli, "Fingerprint recognition," in *Handbook of Biometrics*, A. K. Jain, P. Flynn, and A. A. Ross, Eds. Boston, MA, USA: Springer, 2008, pp. 23–42.

[3] R. Cappelli, "Unveiling the power of simplicity: Two remarkably effective methods for fingerprint segmentation," *IEEE Access*, vol. 11, pp. 144530–144544, 2023.

[4] R. Cappelli, "Exploring the power of simplicity: A new state-of-the-art in fingerprint orientation field estimation," *IEEE Access*, vol. 12, pp. 55998–56018, 2024.

[5] T. Dalzell, *The Routledge Dictionary of Modern American Slang and Unconventional English*. Evanston, IL, USA: Routledge, 2009.

[6] D. F. Alwin and B. A. Beattie, "The KISS principle in survey design: Question length and data quality," *Sociol. Methodol.*, vol. 46, no. 1, pp. 121–152, Aug. 2016.

[7] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777–789, Aug. 1998.

[8] X. Jiang, "Fingerprint image ridge frequency estimation by higher order spectrum," in *Proc. Int. Conf. Image Process.*, vol. 1, Sep. 2000, pp. 462–465.

[9] J. Yang, L. Liu, T. Jiang, and Y. Fan, "A modified Gabor filter design method for fingerprint image enhancement," *Pattern Recognit. Lett.*, vol. 24, no. 12, pp. 1805–1817, Aug. 2003.

[10] T. Orczyk and L. Wieclaw, "Fingerprint ridges frequency," in *Proc. 3rd World Congr. Nature Biologically Inspired Comput.*, Oct. 2011, pp. 558–561.

[11] C. Gottschlich, "Curved-region-based ridge frequency estimation and curved Gabor filters for fingerprint image enhancement," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2220–2227, Apr. 2012.

[12] D. Maio and D. Maltoni, "Ridge-line density estimation in digital images," in *Proc. 14th Int. Conf. Pattern Recognit.*, vol. 1, Aug. 1998, pp. 534–538.

[13] A. Almansa and T. Lindeberg, "Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection," *IEEE Trans. Image Process.*, vol. 9, no. 12, pp. 2027–2042, Dec. 2000.

[14] R. Lehtihet, W. E. Oraiby, and M. Benmohammed, "Ridge frequency estimation for low-quality fingerprint images enhancement using Delaunay triangulation," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 28, no. 1, Feb. 2014, Art. no. 1456002.

[15] S. Chikkerur, V. Govindaraju, and A. N. Cartwright, "Fingerprint image enhancement using STFT analysis," in *Pattern Recognition and Image Analysis*. Berlin, Germany: Springer, 2005.

[16] V.-V. Patriciu and S. Spinu, "Fingerprint ridge frequency estimation in the Fourier domain," *Adv. Electr. Comput. Eng.*, vol. 14, no. 4, pp. 95–98, 2014.

[17] Z. M. Kovács-Vajna, R. Rovatti, and M. Frazzoni, "Fingerprint ridge distance computation methodologies," *Pattern Recognit.*, vol. 33, no. 1, pp. 69–80, Jan. 2000.

[18] X. Zhan, Z. Sun, Y. Yin, and Y. Chu, "Fingerprint ridge distance estimation: Algorithms and the performance," in *Advances in Biometrics*. Berlin, Germany: Springer, 2005.

[19] D. H. Thai, S. Huckemann, and C. Gottschlich, "Filter design and performance evaluation for fingerprint image segmentation," *PLoS ONE*, vol. 11, no. 5, May 2016, Art. no. e0154160.

[20] B. Dorizzi, R. Cappelli, M. Ferrara, D. Maio, D. Maltoni, N. Houmani, S. Garcia-Salicetti, and A. Mayoue, "Fingerprint and on-line signature verification competitions at ICB 2009," in *Advances in Biometrics*. Berlin, Germany: Springer, 2009.

[21] (2024). *FVC-onGoing: On-Line Evaluation of Fingerprint Recognition Algorithms*. Accessed: Feb. 26, 2024. [Online]. Available: https://biolab.csr.unibo.it/fvcongoing

[22] J. Feng, J. Zhou, and A. K. Jain, "Orientation field estimation for latent fingerprint enhancement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 925–940, Apr. 2013.

[23] Y. Tang, F. Gao, J. Feng, and Y. Liu, "FingerNet: An unified deep network for fingerprint minutiae extraction," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 108–116.

[24] Y. Zhu, X. Yin, and J. Hu, "FingerGAN: A constrained fingerprint generation scheme for latent fingerprint enhancement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8358–8371, Jul. 2023.

[25] S. Kriangkhajorn, K. Horapong, and V. Areekul, "Spectral filter predictor for progressive latent fingerprint restoration," *IEEE Access*, vol. 12, pp. 66773–66800, 2024.

[26] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-Stokes, fluid dynamics, and image and video inpainting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, p. 1.

[27] *OpenCV: Open Source Computer Vision Library*. Accessed: Oct. 18, 2024. [Online]. Available: https://opencv.org/

[28] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2002: Second fingerprint verification competition," in *Proc. Int. Conf. Pattern Recognit.*, Aug. 2002, pp. 811–814.

[29] R. Cappelli, D. Maio, D. Maltoni, J. L. Wayman, and A. K. Jain, "Performance evaluation of fingerprint verification systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 3–18, Jan. 2006.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*. Cham, Switzerland: Springer, 2015.

[32] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Cham, Switzerland, 2018, pp. 801–818.

[33] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11966–11976.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017, *arXiv:1412.6980*.

[35] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. 4th Int. Conf. Learn. Represent.*, 2016, pp. 1–4.

[36] *Keras*. Accessed: Oct. 18, 2024. [Online]. Available: https://keras.io/


**RAFFAELE CAPPELLI** received the degree in computer science from the University of Bologna, in 1998, and the Ph.D. degree, in 2002.

He is currently an Associate Professor with the Department of Computer Science and Engineering, University of Bologna, and a member of the Biometric System Laboratory, Cesena. He is one of the organizers of the Fingerprint Verification Competitions (FVC 2000–FVC 2006 and FVC-onGoing) and one of the authors of the synthetic fingerprint generator (SFinGe).

Open Access funding provided by 'Alma Mater Studiorum - Università di Bologna' within the CRUI CARE Agreement