September 10, 2019

Fingerprint-Matching Algorithm Using Polar Shapelets

Firas Sammoura

Jean-Marie Bussat

Fingerprint-Matching Algorithm Using Polar Shapelets

Abstract:

An image, such as a fingerprint, can be decomposed into a linear combination of polar shapelet-base functions. This publication describes a fingerprint-matching algorithm that uses polar shapelet-base functions. Using polar shapelet-base functions, a fingerprint image block can be separated into components with explicit rotational symmetries. Polar shapelet-base functions can represent the fingerprint image through compact parameterization or encoder representation due to their interpretation in terms of the rotational angle θ, and due to their separability by a distance r. Therefore, the use of polar shapelets enables a convenient and robust method to perform fingerprint image manipulation, analysis, and matching. Polar shapelet-base functions are special types of steerable filters that have rotational symmetry. When the fingerprint image is convolved with a polar shapelet-base function, the magnitude of the convolution output is rotationally invariant, and the relative rotation between two fingerprint images is the phase shift in the convolution output, which enables calculating the rotation angle between two matching image blocks with relative ease. Polar shapelet-base functions can be utilized to create a machine-learned (ML) model that is composed of harmonic and rotationally symmetric convolution filters. The fingerprint-matching algorithm pre-specifies the rotation order of each filter, but the size and shape of the convolution filter is optimized using the ML model. Also, the fingerprint-matching algorithm optimizes a TensorFlow implementation for each convolution filter in the radial direction r. The ML model determines an optimized set of filters that can increase the matching between two rotated images. The described fingerprint-matching algorithm offers high-resolution fingerprint images, low computation latency, low image energy residuals, and high matching rates.

Keywords:

Fingerprint, iris scan, palmprint, infant footprint, biometric, authentication, fingerprint-matching algorithm, feature-matching algorithm, image encoding, image decomposition, polar shapelet, polar shapelet function, polar shapelet-base functions, Laguerre polynomial, explicit rotational symmetry, image block, image convolution, machine learning, ML, artificial intelligence, AI.

Background:

Since antiquity, various civilizations have used fingerprints to identify individuals because human fingerprints are detailed, nearly unique, difficult to alter, and durable over the life of an individual. These characteristics make fingerprints suitable as long-term markers of human identity. In modern times, virtually all countries use fingerprint recognition, in some form or another, to identify some of their citizens. Nevertheless, fingerprint recognition is more than a government or a legal tool for identifying alive or deceased individuals—many user equipment (UE) give a user a choice to use fingerprint identification to gain access to a user device, door entrance, vault, application software, physical locations, and/or virtual activities that the user wants guarded. Figure 1 illustrates an example of a fingerprint image.

Figure 1

The analysis of fingerprints for matching purposes generally requires the comparison of patterns and minutiae. The three main patterns of the fingerprint ridges are the arch, the loop, and the whorl. An arch is a fingerprint ridge that enters from one side of the finger, rises in the center forming an arc, and then exits the other side of the finger. A loop is a fingerprint ridge that enters from one side of the finger, forms a curve, and then exits on that same side of the finger. A whorl is a fingerprint ridge that is circular around a central point. On the other hand, the minutiae are features of fingerprint ridges, such as ridge ending, bifurcation, double bifurcation, trifurcation, short (independent) ridge, island, lake (ridge) enclosure, spur, bridge, delta, core, and so forth, as illustrated in Figure 2.

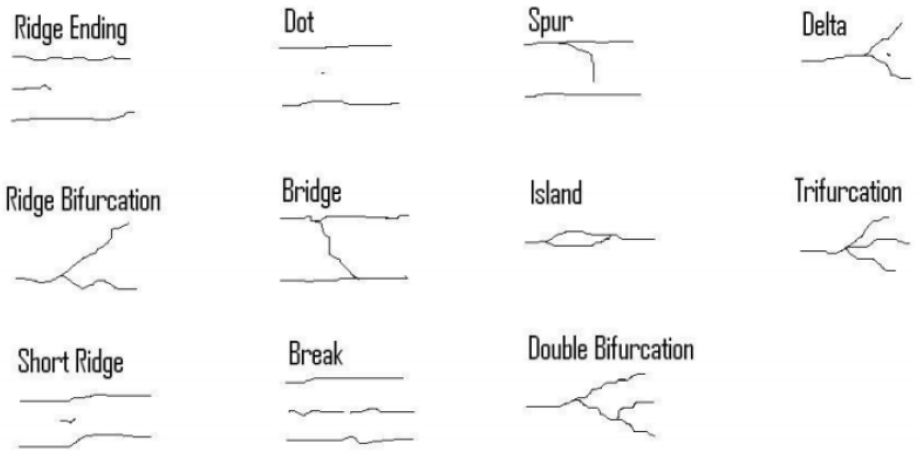


Figure 2

The UE may use a fingerprint sensor to capture a fingerprint image. When the UE uses a large fingerprint sensor, minutiae matching is achieved with high success. Nevertheless, some UE (*e.g.*, smartphones) use small fingerprint sensors, which decrease the number of minutiae being scanned. The UE struggles to make a positive minutiae-match when it can only scan a small number of minutiae.

The small fingerprint sensor is one of the reasons why many UE manufacturers use pattern-correlation matching, and they often try to correlate the full fingerprint. First, the UE tries to match

the alignment and the orientation of the fingerprint, then the UE correlates the full fingerprint image. This technique, however, cannot easily support 1000 dots-per-inch (DPI) resolution fingerprint images. A higher-resolution image can increase the fingerprint-matching success rate. To increase the resolution of a fingerprint, an original device manufacturer (ODM) can utilize a fusion of minutiae and pattern-correlation matching, which can substantially increase the matching success rate and enables an increase in computation speed by evaluating only part of the full fingerprint. The fusion of minutiae and pattern-correlation matching requires performing a polar representation of a meshgrid, a two-dimensional (2D) Fast Fourier Transform (FFT) of the polar representation, and a pointwise Fourier multiplication. These calculations, however, have some inherent computational latency.

Therefore, it is desirable to have a technological solution that can increase the computation speed of a high-resolution fingerprint image.

Description:

An image, such as a fingerprint, may be decomposed into a linear combination of shapelet-base functions. In mathematics, a shapelet is an orthogonal (orthonormal) polynomial (*e.g.*, *Laguerre, Hermite, Jacobi*) that is weighted (multiplied) by a *Gaussian* function (*Gaussian*). This publication describes a fingerprint-matching algorithm that uses polar shapelet-base functions.

In a shapelet formalism, an image can be decomposed into a weighted sum of orthogonal-base functions. Thus, using polar shapelet-base functions, a fingerprint image block can be separated into components with explicit rotational symmetries, as is illustrated in Figure 3.

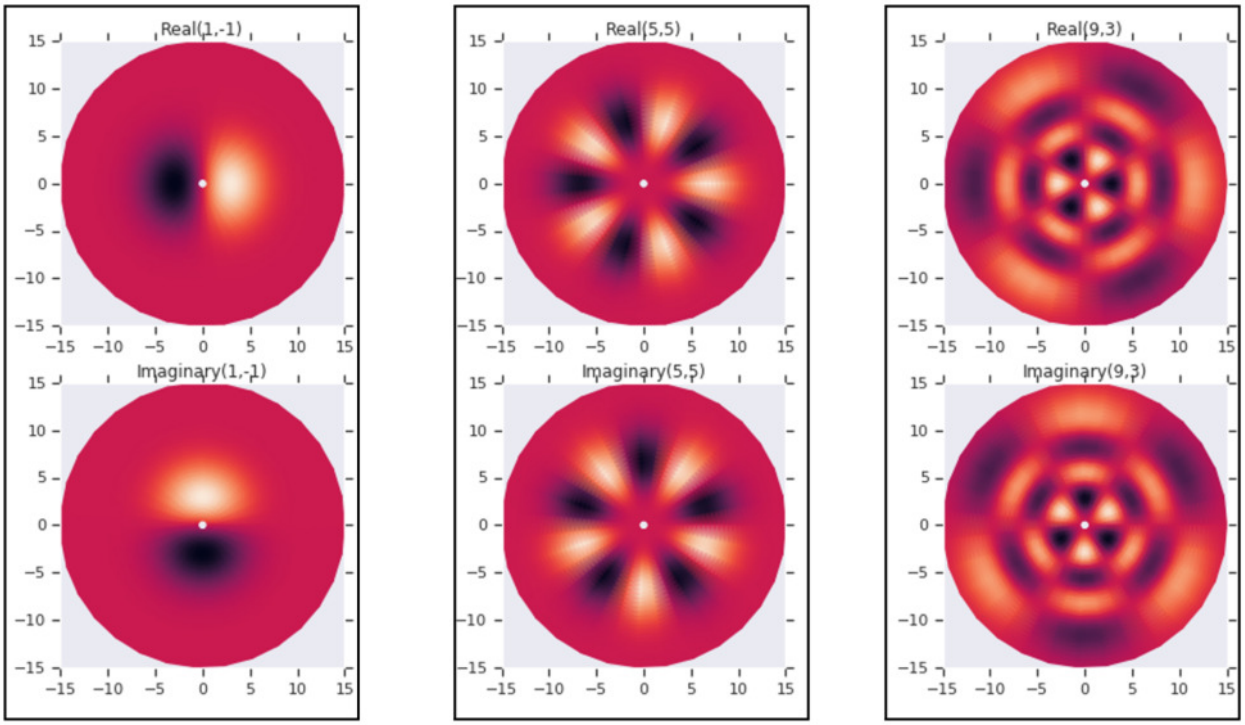


Figure 3

Figure 3 illustrates three examples of real (upper row) and imaginary (bottom row) components of three polar shapelet-base functions with different order (*n, m*) numbers. Figure 3A illustrates a (1, 1) order polar shapelet-base function, Figure 3B illustrates a (5, 5) order polar-shapelet function, and Figure 3C illustrates a (9, 3) order polar shapelet-base function.

Using polar shapelet-base functions enables technical professionals (*e.g.*, engineers, mathematicians, scientists) to represent a fingerprint image through compact parameterization or encoder representation due to their interpretation in terms of the rotational angle θ, and due to their separability by a distance *r*, as is illustrated in Figure 3. Therefore, the use of polar shapelets enables a convenient and robust method to perform fingerprint image manipulation, analysis, and matching.

The polar shapelet-base function *X* can be constructed using associated *Laguerre* polynomials *L* that are weighted by a *Gaussian* and a given scale β, as is defined in Equation 1:

$$\mathrm{X}_{n,m}(r,\theta,\beta) = \frac{(-1)^{\frac{n-|m|}{2}}}{\beta}\left[\frac{\left(\frac{n-|m|}{2}\right)!}{\pi\left(\frac{n+|m|}{2}\right)}\right]^{\frac{1}{2}}\left(\frac{r}{\beta}\right)^{2+|m|} L_{\frac{n-|m|}{2}}^{|m|} e^{\frac{-r^2}{2\beta^2}} e^{-im\theta} \qquad \textbf{(1)}$$

where *n* is a non-negative integer, and *m* is an integer between -*n* and *n*, in steps of two (2). For example, if *n* = 1, *m* = -1, 1. As another example, if *n* = 5, *m* = -5, -3, -1, 1, 3, 5.

The associated *Laguerre* polynomial is defined in Equation 2:

$$L_p^q(x) = \frac{x^{-q}e^x}{p!}\frac{d^p}{dx^p}(x^{p+q}e^{-x}) \qquad \textbf{(2)}$$

A smooth function *f* in polar coordinates can be decomposed into a set of equations defined in Equation 3 and Equation 4:

$$\begin{cases} f(r,\theta) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} f_{n,m}\mathrm{X}_{n,m}(r,\theta,\beta) & \textbf{(3)} \\ f_{n,m} = \iint_R f(r,\theta)\,\mathrm{X}_{n,m}^*(r,\theta,\beta) r dr d\theta & \textbf{(4)} \end{cases}$$

where *R* represents the area of mathematical integration.

Recall that polar shapelet-base functions are orthogonal. Thus, they can be defined using Equation 5 and Equation 6:

$$\iint_R \mathrm{X}_{n,m}^*(r,\theta,\beta)\, X_{n',m'}(r,\theta,\beta) r dr d\theta = \delta_{n,n'}\delta_{m,m'} \qquad \textbf{(5)}$$

where * is the mathematical representation of the complex conjugate. Note that *X(n,m)* is the polar-shapelet function of mode *(n, m)*, while *X(n', m')* is the polar-shapelet function of mode *(n', m')*.

$$\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\mathrm{X}_{n,m}(r,\theta,\beta)\mathrm{X}_{n,m}(r',\theta',\beta) r dr d\theta = \delta(r-r')\delta(\theta-\theta') \qquad \textbf{(6)}$$

where r' refers to a new distance *r*, and θ' refers to a new rotational angle θ.

In general, polar shapelet-base functions have real and imaginary components (complex components), as is illustrated in Figure 4.

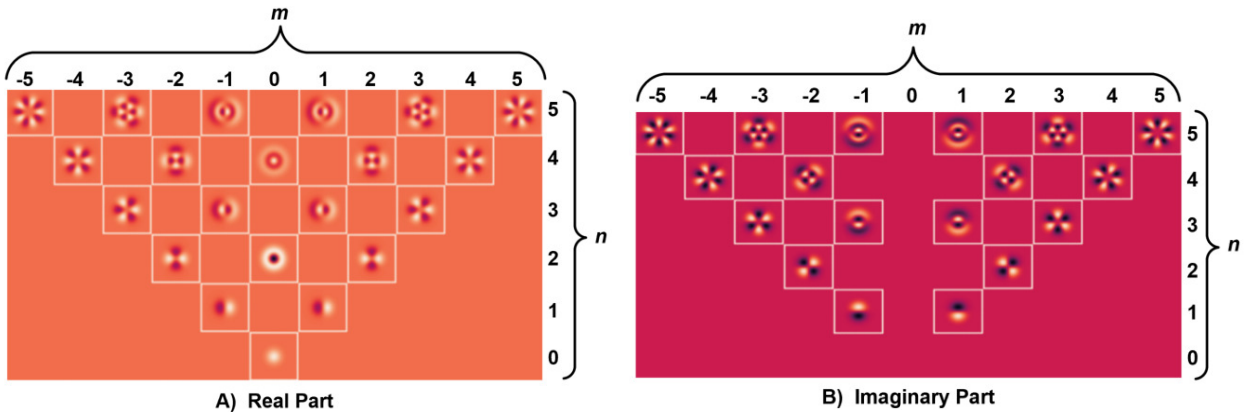


Figure 4

Recall that n is a non-negative integer (such as 0, 1, 2, …,∞), and m is an integer between $-n$ and n in steps of two (2) integers (such as $-n+2$, $-n+4$,…,-5, -3, -1, 0, 1, 3, 5,…, $n-4$, $n-2$). The base functions of order (n, m) and (n, $-m$) are complex conjugates and, as such, are dependent on each other. Therefore, in general, the polar shapelet-base functions have real (Figure 4A) and imaginary (Figure 4B) components. It is worth noting that polar shapelet-base functions with m = 0 are wholly real (not complex) and do not have an imaginary part, as is illustrated in Figure 4B.

An important and helpful property of polar shapelet-base functions is filter steerability, as is illustrated in Figure 5.

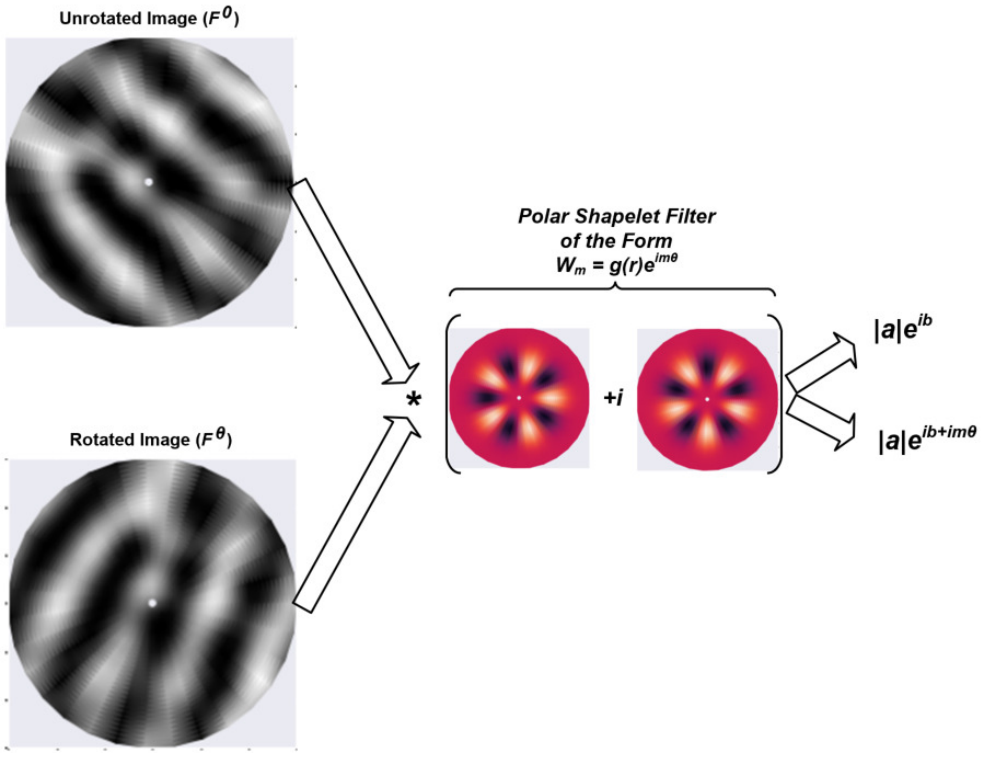


Figure 5

Polar shapelet-base functions are special types of steerable filters that have rotational symmetry. The convolution of a rotated fingerprint image F^{θ} with a polar shapelet $X_{n,\,m}$ of the order $(n,\,m)$ is equal to the convolution of the unrotated version of the fingerprint image F^{0} with the same polar shapelet $X_{n,\,m}$ and shifted by a phase of $m\theta$, as is defined in Equation 7:

$$W_m * F^{\theta} = e^{\mathrm{im}\theta} \cdot (W_m * F^{0}) \qquad \textbf{(7)}$$

where (*) depicts the mathematical operation of convolution, and (·) depicts the mathematical operation of multiplication.

Consequently, the magnitude of the convolution output is rotationally invariant. In addition, the relative rotation between two fingerprint images is the phase shift in the convolution output using polar shapelets of the order m = -1, which enables calculating the rotation angle between two matching blocks with relative ease.

The use of polar shapelet-base functions and their properties enable the decomposition and reconstruction of a fingerprint image without having to take an infinite number of weighted sums, as is illustrated in Figure 6A, Figure 6B, Figure 6C, and Figure 6D.

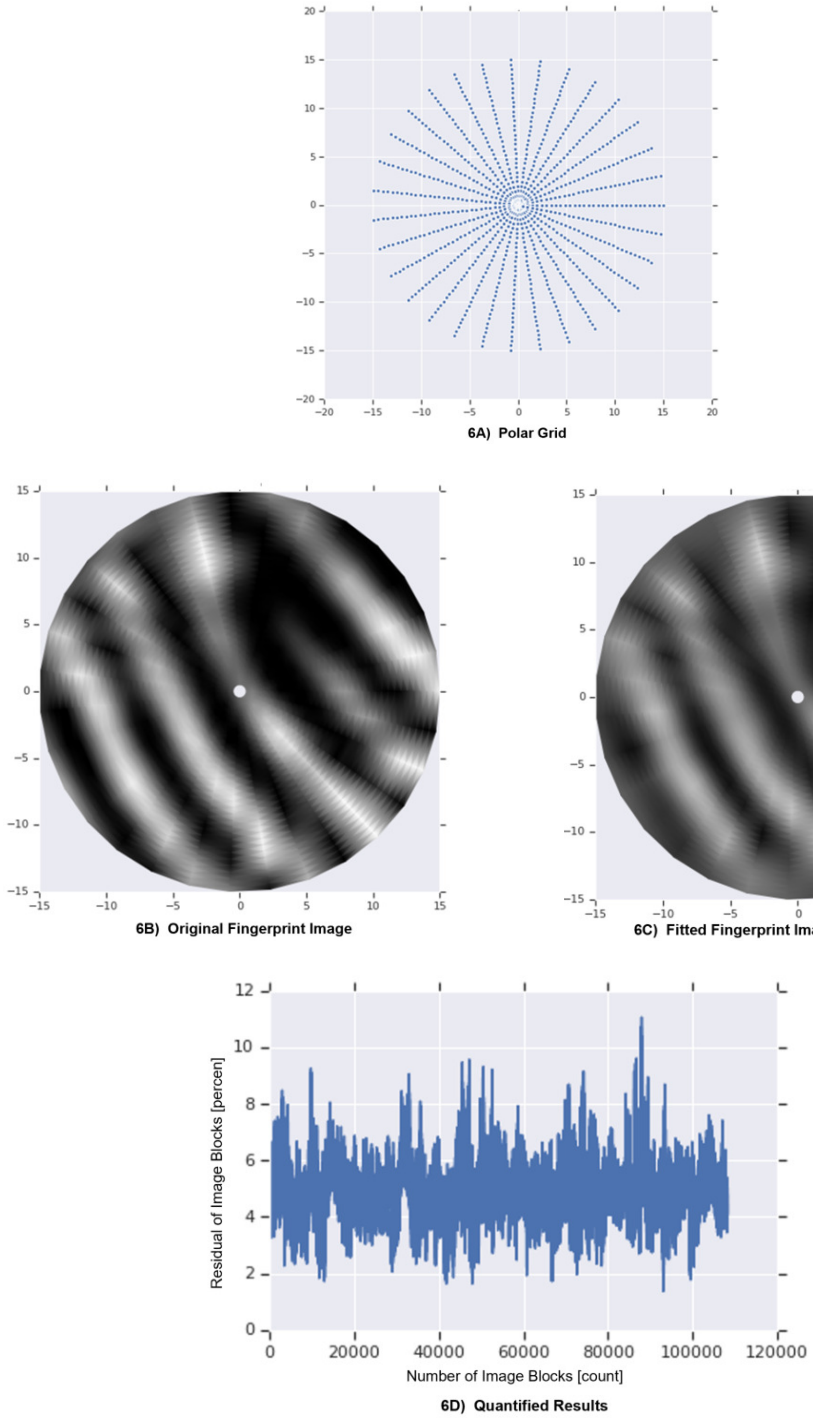


Figure 6

Instead of using an infinite number of weighted sums, the fingerprint-matching algorithm, using polar shapelet-base functions, can truncate a series of vectors; in the example of Figure 6, to an order of n=17. Using an order of n=17, the first 171 base vectors can be calculated, and a polar shapelet-base matrix can be formed. Recall that if n=17, there are 171 combinations of (n, m) vectors, and only 90 vectors are independent because the rest are complex conjugate vectors of other vectors. The complex conjugate vectors, however, are still used to decompose and reconstruct a fingerprint image.

A fingerprint image block of 31x31 pixels is implemented. As such, each base vector has 961 (31x31=961) components calculated at the points of a polar grid that are determined using:

- Line space grid of 31 steps for the angular direction θ from 0 to 2π (Figure 6A); and
- Line space grid of 31 steps for the radial direction r from a distance of 0.5 to 15 (Figure 6A, Figure 6B, and Figure 6C).

The decomposition of a fingerprint-image block into polar shapelet-base functions uses numerical integration, where the coefficient $f_{n,m}$ is calculated using Equation 4, and a matrix of the form $X^*(r_i, \theta_j)r(r_i, \theta_j)\Delta r \Delta\theta$ is created. Note that the calculation and use of $f_{n,m}$ enables discretizing the polar shapelet-base function. Then, $f(r_i, \theta_j)$ is multiplied by the matrix to find the respective coefficients. Figure 6C illustrates qualitative (by visual inspection) results of these calculations, and Figure 6D illustrates quantitative results of these calculations. Figure 6B illustrates the original fingerprint image, and Figure 6C illustrates a reconstructed fingerprint image using the 171 vectors. Figure 6D illustrates the percent residual of image blocks for 111,000 image blocks. The maximum residual of the 111,000 image blocks is about eleven (11) percent, and the average residual of the 111,000 image blocks is about five (5) percent.

Therefore, using only 171 vectors can enable the fingerprint-matching algorithm to decompose and reconstruct high-resolution fingerprint images, as is further illustrated in Figure 7.

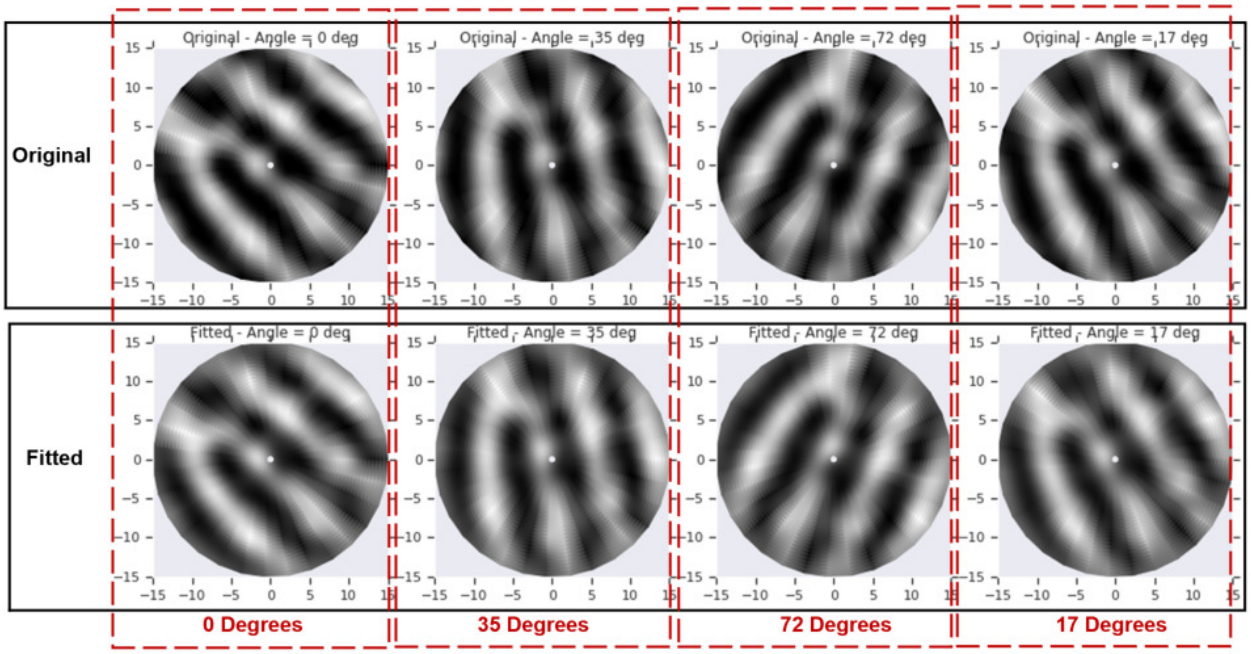


Figure 7

The upper row of Figure 7 illustrates four examples of an original fingerprint image that is rotated by zero (0) degrees (unrotated), 35 degrees, 72 degrees, and 17 degrees, respectively from left to right. The bottom row of Figure 7 illustrates four examples of a fitted (predicted using the algorithm) fingerprint image that is rotated by zero (0) degrees (unrotated), 35 degrees, 72 degrees, and 17 degrees, respectively from left to right. Qualitatively (by visual inspection), Figure 7 illustrates that the fitted image retains most of the image information (image energy) of the original image regardless of the degree of rotation. More specially, the calculations quantize the error between the actual rotation and the predicted rotation, as follows:

- Actual rotation is 35 degrees, while predicted rotation 35.3 degrees;
- Actual rotation is 72 degrees, while predicted rotation is 72.6 degrees; and
- Actual rotation is 17 degrees, while predicted rotation is 17.1 degrees.

As supported by Equation 1, Equation 3, Equation 4, Equation 5, and Equation 6, the scale size β plays a role in polar shapelet-base functions, as is illustrated in Figure 8.

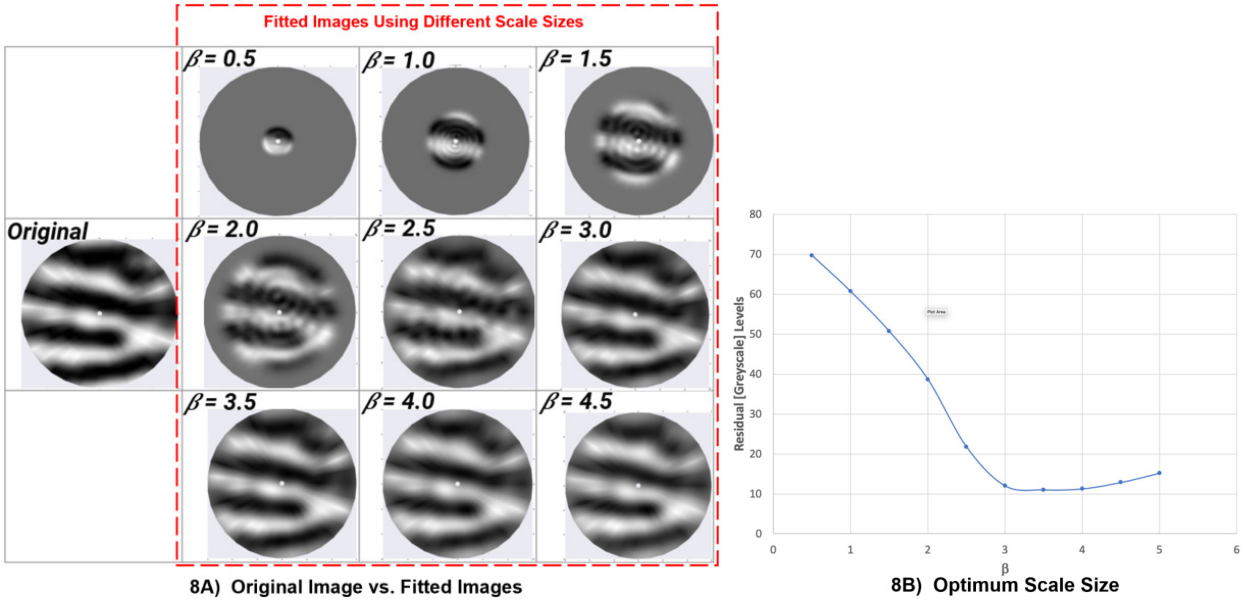
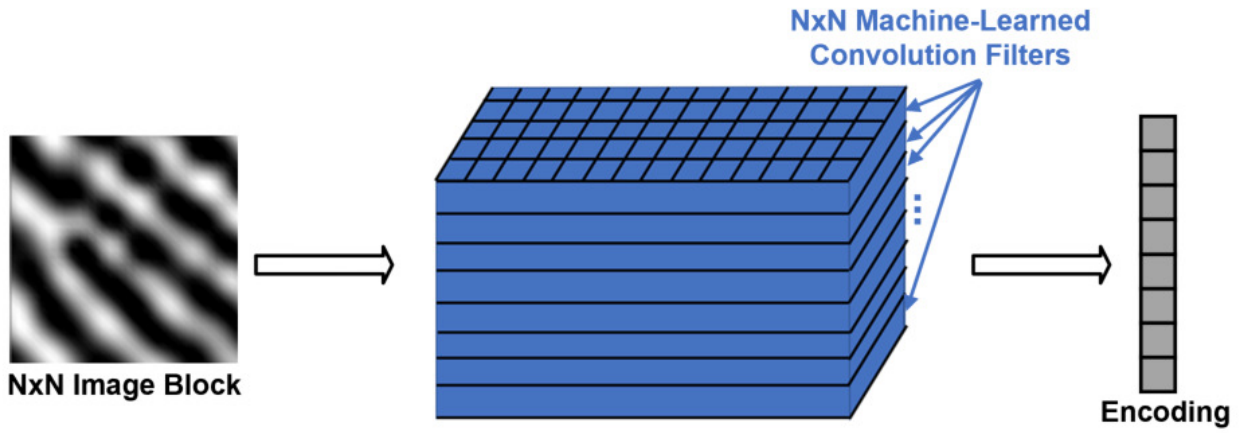


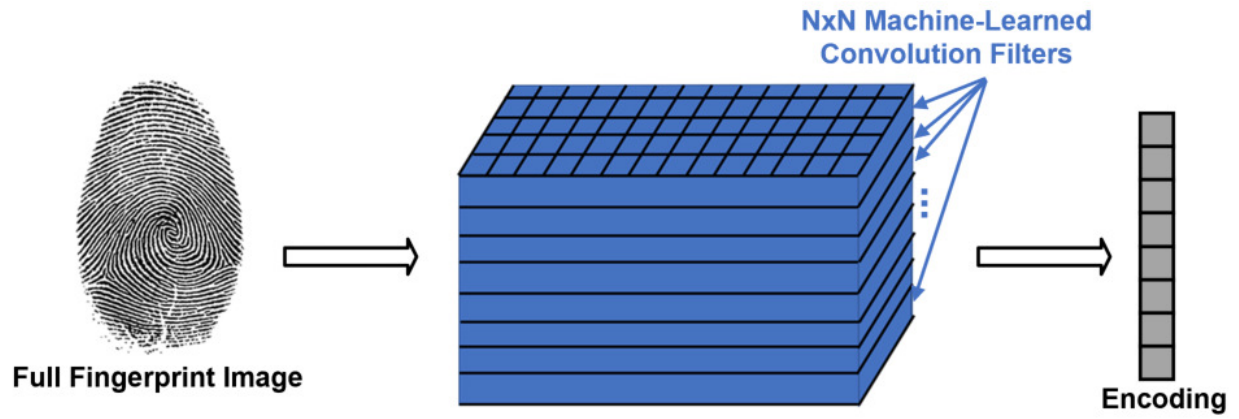
Figure 8

Figure 8A illustrates a comparison between an original fingerprint image versus a fitted fingerprint image block with $n_{max} = 17$, using different scale sizes β; specifically, $\beta = 0.5$, 1.0, 1.5, 3.0, 2.5, 3.0, 3.5, 4.0, and 4.5. When using a scale size of $\beta = 0.5$, the residual (grayscale level) is exceedingly high, producing a poor fitted image. Qualitatively (by visual inspection), Figure 8A illustrates that the fitted images with scale sizes of $\beta = 2.5$, 3.0, 3.5, 4.0, and 4.5 have more similar image energies to the original fingerprint image compared to the fitted images with scale sizes of $\beta = 0.5$, 1.0, 1.5, and 2.0. Quantitively, Figure 8B illustrates that there is an optimum scale size β that produces the best fitted image; in the example of Figure 8, a scale size β between 3.0 and 4.0. Therefore, ever-increasing the scale size β to produce a fitted image does not necessarily yield a better fitted image because beyond an optimum scale size β, mathematical integration errors hinder image decomposition and reconstruction.

The versatility of the polar shapelet-base functions and their properties can be utilized to train a machine-learned (ML) model that increases the quality of a fingerprint image, as is illustrated in Figure 9A and Figure 9B.



9A) Using a Machine-Learned Model to Encode an NxN Image Block



9B) Using a Machine-Learned Model to Encode a Full Fingerprint Image

Figure 9

Figure 9A illustrates an NxN image block that is input to the ML model that uses a stack of NxN convolution filters to create an encoding of the NxN image block. Recall from Figure 6 that the encoded image of a 31x31 block represents only 90 (instead of 961) features, making the image encoding efficient. To produce the fingerprint image, the fingerprint-matching algorithm

decodes the encoded image (not shown in Figure 9). The ML model may be a standard neural network-based model with corresponding layers required for processing input features, such as fixed side vectors, image energies, rotation angles, variable length sequences, and so forth. In addition, the ML model may be a support vector machine, a recurrent neural network, a convolutional neural network, a dense neural network, heuristics, or a combination thereof. The decoded image and its residual can be fed back to the ML model to make better future calculations.

The convolution filters are harmonic, and they are rotationally symmetric. The fingerprint-matching algorithm pre-specifies the rotation order of each filter, but the size and shape of the convolution filter is optimized using the ML model. Also, the fingerprint-matching algorithm optimizes a TensorFlow implementation for each convolution filter in the radial direction *r*. The implemented harmonic network determines an optimized set of filters that can increase the matching between two rotated images. For small or partial fingerprint image blocks, as is illustrated in Figure 9A, the convolution of each filter with the image produces a single output. For large image blocks (such as a full fingerprint), however, the convolution of each filter with the image produces a matrix output, as is illustrated in Figure 9B, which increases the computation speed of image encoding. Both single and matrix outputs follow the properties of harmonic filters. The fingerprint-matching algorithm can treat a single and/or a matrix output similarly, by using matrix multiplication.

Given the large computational power that machine-learning can use to train a model, the model training can be performed on a cloud, server, or other capable computing device or system. Periodic model updates are sent to a user's computing device, which allows the user's computing device to execute the ML model even if that device does not have the resources to update the model

itself. Instead or in addition, some or all of the model training can be performed on the user's computing device.

This algorithm can be scaled to handle higher than 1000-DPI resolution images and large area fingerprint images. In addition, this algorithm can be used for other forms of biometric matching, such as iris, palmprint, and baby footprint. In that aspect, this algorithm is versatile.

Further to the descriptions above, a user may be provided with controls allowing the user to make an election as to both if and when systems, programs, or features described herein may enable collection of user information (*e.g.*, biometric information, a user's preferences, a user's current location), and if the user is sent content or communications from a server. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, so that a particular location of a user cannot be determined. Thus, the user may have control over what information is collected about the user, how that information is used, and what information is provided to the user. The user may also choose to disable the fingerprint-matching algorithms from their device.

In summary, the described fingerprint-matching algorithm, which uses polar shapelets, offers high-resolution fingerprint images, low computation latency, low image energy residuals, and high matching rates.

References:

[1] Patent Publication: US20180137329A1. User authentication method using fingerprint image and method of generating coded model for user authentication. Filing Date: July 19, 2017.

[2] Patent Publication: WO2014004210A1. Fingerprint enrollment using partial fingerprints. Filing Date: June 19, 2013.

[3] Sammoura, Firas, Kuntal Sengupta; and Jean-Marie Bussat. "Fingerprint-Matching Algorithms." *Technical Disclosure Commons* (May 29, 2019). https://www.tdcommons.org/dpubs_series/2228.