

# Correlated Feature Selection Model based on Swarm Intelligence to Enhance Rotational Invariant Fingerprint Authentication using Firefly Optimized Algorithm

Adil Al-Azzawi<sup>1</sup>

ECE Dept., University of Missouri-Columbia, Columbia, MO, USA <sup>1</sup>

**Abstract:** Fingerprint characteristic of every human are unique and stable, therefore widely accepted for personal identification. Accurate and reliable fingerprint recognition is a challenging task, which heavily depends on the quality of the fingerprint images. It is well-known that the fingerprint recognition systems are very sensitive to noise and other image transformations such as rotation. In this paper an enhanced algorithm is introduced to make the matching process rotational invariant. The proposed algorithm is robust to match two identical fingerprint images which are spatially aligned at different rotational angles and gives better FMR and FNMR ratio for images rotated at different angles. A Biometric Fingerprint authentication algorithm using firefly algorithm (FA) based feature selection approach which we named firefly optimization algorithm (FOA) is proposed to address the above issues. The search in (FOA) is iteratively guided by a fitness function defined to maximize class separation to identify new features instead of the traditional Minutiae (Termination and Bifurcation features). The first contribution is the formulation of a new feature selection algorithm for fingerprint recognition based on the DWT algorithm, which solves the localization problem by applying (FOA) separately to four sub-bands to increase the recognition rate and to speed up feature selection. The second contribution is an invariant moment matching algorithm which is proposed as a matching algorithm to address some misclassified features and to increase the matching accuracy. The proposed algorithm was found to generate excellent recognition results which admit 100% accuracy when applied on the FCV2002, 2004 and 2006 dataset. Also, it admits accuracy ranged 96.35-100% when applied to fingerprints with a rotation of 0-360°.

**Keywords:** Authentication system, Swarm Intelligence, PSO, Invariant moment matching algorithm.

## I. INTRODUCTION

Recognition of a person by means of biometric characteristics is an emerging technology. Among the possible biometric traits like face, iris, speech, and hand geometry, fingerprint is the most widely used trait, because of its distinctiveness and persistence over time [1]. A fingerprint image is a pattern of ridges and valleys, with ridges as dark lines while valleys as light areas between the ridges. Ridges and valleys generally run parallel to each other, and their patterns can be analyzed on a global and local level [2]. Global analysis of the fingerprint image is done to extract singular regions like loop, delta, and whorl. Many matching algorithms use the center of the highest loop type singularity, known as the core, to pre-align fingerprint images for better results. These singularities help form the 5 major classes [3] of fingerprints.

The noise and other distortion during the acquisition of the fingerprint and errors in the minutia extraction process can result in spurious and missing minutiae that easily degrade the performance of the recognition [4]. Another problem is that the rotation and displacement of the finger placed on the sensor, can lead to different images for the same fingerprint as and that what we want to focus on in this paper and as it shown in Fig.1 such that they have only a

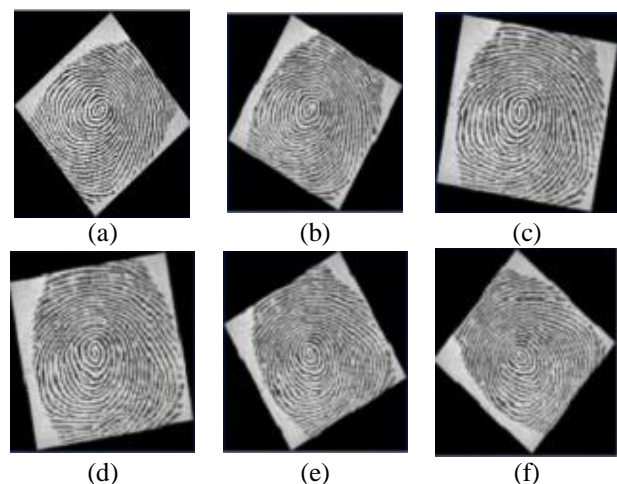


Fig. 1. Input image rotated at angles (a) -50, (b) -30, (c) -10, (d) 10, (e) 30, (f) 50.

partial overlap area resulting in only a small number of corresponding minutiae points [5]. The problems with minutiae extraction can be more severe if the fingerprint is acquired using a compact solid-state sensor. They provide only a small contact area for the fingertip and, therefore,

capture only a limited portion of the fingerprint pattern as it shown in Fig.2 (b) and (c) [6].

Metaheuristic algorithms form an important part of contemporary global optimization algorithms, computational intelligence and soft computing. These algorithms are usually nature-inspired with multiple interacting agents. A subset of metaheuristics is often referred to as swarm intelligence (SI) based algorithms, and these SI-based algorithms have been developed by mimicking the so-called swarm intelligence characteristics of biological agents such as birds, fish, humans and others. For example, particle swarm optimization was based on the swarming behavior of birds and fish [7], while the firefly algorithm was based on the flashing pattern of tropical fireflies [8] [9] and cuckoo search algorithm was inspired by the brood parasitism of some cuckoo species [10]. In the last two decades, more than a dozen new algorithms such as particle swarm optimization, differential evolution, bat algorithm, firefly algorithm and cuckoo search have appeared and they have shown great potential in solving tough engineering optimization problems [11] [12] [13] [14] [15] [16]. Among these new algorithms, it has been shown that firefly algorithm is very efficient in dealing with multimodal, global optimization problems. In this paper, we will first outline the fundamentals of Firefly Algorithm (FA), and then review the latest developments concerning FA and its variants. We also highlight the reasons why FA is so efficient. Furthermore, as the balance of exploration and exploitation is important to all metaheuristic algorithms, we will then discuss the optimality related to search landscape and algorithms. Using the intermittent search strategy and numerical experiments, we show that our Firefly Optimized Algorithm (FOA) is significantly more efficient than intermittent search strategy.

The rest of the paper is organized as follows. Section II discusses some related work and highlights how our approach differs. Section III introduces our proposed system. Section IV shows some experimental results that compare our works with the state-of-the-art approaches, and Section V makes the conclusion.

## II. RELATED WORKS

Most of the previous works for fingerprint recognition use the minutiae features as the fingerprint features. We summarize some related studies below.

Ross et al. [17] describe a hybrid approach that combines a set of minutiae with a ridge feature map extracted through a set of Gabor filters. The energy of the filtered image, which is comprised of the ridge feature map and a set of minutiae pairs, is used for evaluating the similarity between test and trainee fingerprint images. Tico et al. [18] proposed a method for fingerprint recognition based on local texture features extracted from the wavelet transform of a discrete image. The ROI of  $64 \times 64$  pixels is cropped around the core point (detected manually). It is then divided into four non-overlapping blocks that are each of  $32 \times 32$  in size and a set of wavelets up to level 4 are applied to each block. At each

level the standard deviation of the wavelet coefficients is computed over 48 blocks to get a feature vector length of 48 for each fingerprint. The similarity is calculated using the Euclidian distance between the feature vectors of the test and the trained fingerprint images. Amornraksa and Tachaphetpiboon [19] have also proposed a local texture analysis method based on the discrete cosine transform (DCT) for fingerprint matching. This method involves steps that are almost like the ones used by Tico's method, except that the feature vector length is six per block (or a total length of 24 for four sub-images) instead of 48. However, the two features which are not robust to rotation. Jin et al. [20], using the integrated wavelets and Fourier-Mellin transform (WFMT), developed a transform-based descriptor. This method is not vulnerable to rotation, transformation, and shape distortion. Multiple WFMTs are used to form a reference WFMT feature to reduce the variability of the fingerprint images. But this makes the matching process time-consuming. Ravinder Kuma [21] develops a robust fingerprint matching system by extracting the circular region of interest (ROI) of a radius of 50 pixels centered at the core point. Maximizing their orientation correlation aligns two fingerprints that are to be matched. The modified Euclidean distance computed between the extracted orientation features of the sample and query images is used for matching. Extensive experiments were conducted over four benchmark fingerprint datasets of FVC2002 and two other proprietary databases of RFVC 2002 and the AITDB. The experimental results show the superiority of our proposed method over the well-known image-based approaches in the literature.

We observe that the Minutiae features based approach has some critical issues according to the requirements of today's security systems. One is that they are generally time consuming because some of the approaches need to perform some pre-processing such as removing the false features, enhancement like noise removal, and binarization and thinning to extract accurate features as illustrated in Fig.2, (a) and Fig.2, (b). The second one is its sensitivity to geometric transformation of the fingerprint images because it depends on extracting three important parameters which are the  $x$ ,  $y$  locations and the angle value. Fig.2, (b) and Fig.2, (d) show the features extraction using Practical Swarm Optimization (PSO) and Firefly Optimized Algorithm (FA), the gray area on this figure shows the most significant features that the FOA is going to visit.

In this paper, we first focus on the enhancement of the Firefly Algorithm (FA) feature selection by proposing a DWT based multi swarm feature extraction which we named a Firefly Optimized Algorithm (FOA) to diversify the selected features, and addressing some technical issues in the previous PSO and FA based approach such as search start position selection, features reduction, and robustness features selection. We then propose an invariance moment algorithm to solve the PSO feature sequence matching problem. Note that most of the previous works did not discuss about the rotation issue and its effect on the matching performance.

### III. THE PROPOSED FOA-BASED SYSTEM

To address the limitations of prior works discussed in Section II, we propose an efficient and robust PSO based fingerprint authentication system here. PSO provides a fast feature search/extraction algorithm and is robust to rotation. The proposed FOA-based fingerprint recognition system is framework is shown in Fig. 3. And the whole our proposed system in shown in Fig.4. It consists of six major stages: (i) fingerprint acquisition, (ii) Wavelet transform, (iii) FOA for each sub bandto collect different features. (iv) Moment analysis, and (vi) Matching.

### IV. METHODS

#### A. Firefly Algorithm (FA)

Firefly Algorithm (FA) was first developed by Xin-She Yang in late 2007 and 2008 at Cambridge University [8] [9], which was based on the flashing patterns and behavior of fireflies. FA uses the following three idealized rules. The first one is the fireflies are unisex so that one firefly will be attracted to other fireflies regardless of their sex. Second, the attractiveness is proportional to the brightness, and they both decrease as their distance increases. Thus, for any two flashing fireflies, the less bright one will move towards the brighter one. If there is no brighter one than a firefly, it will move randomly. Third, the brightness of a firefly is determined by the landscape of the objective function. As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the variation of attractiveness  $\beta$  with the distance  $r$  by Eq. (1):

$$\beta = \beta_0 e^{-\gamma r^2} \quad (1)$$

where  $\beta_0$  is the attractiveness at  $r = 0$ . The movement of a firefly  $i$  is attracted to another more attractive (brighter) firefly  $j$  is determined by Eq. (2):

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha_t \epsilon_i^t \quad (2)$$

where the second term is due to the attraction. The third term is randomization with  $\alpha_t$  being the randomization parameter, and  $\epsilon_i^t$  is a vector of random numbers drawn from a Gaussian distribution or uniform distribution at time  $t$ . If  $\beta_0 = 0$ , it becomes a simple random walk. On the other hand, if  $\gamma = 0$ , it reduces to a variant of particle swarm optimization [8]. Furthermore, the randomization  $\epsilon_i^t$  can easily be extended to other distributions such as Levy flights [8]. A demo version of firefly algorithm implementation by Xin-She Yang, without Levy flights for simplicity.

As  $\alpha_t$  essentially control the randomness (or, to some extent, the diversity of solutions), we can tune this parameter during iterations so that it can vary with the iteration counter  $t$ . So, a good way to express  $\alpha_t$  is to use Eq. (3):

$$\alpha_t = \alpha_0 \delta^t, \quad 0 < \delta < 1 \quad (3)$$

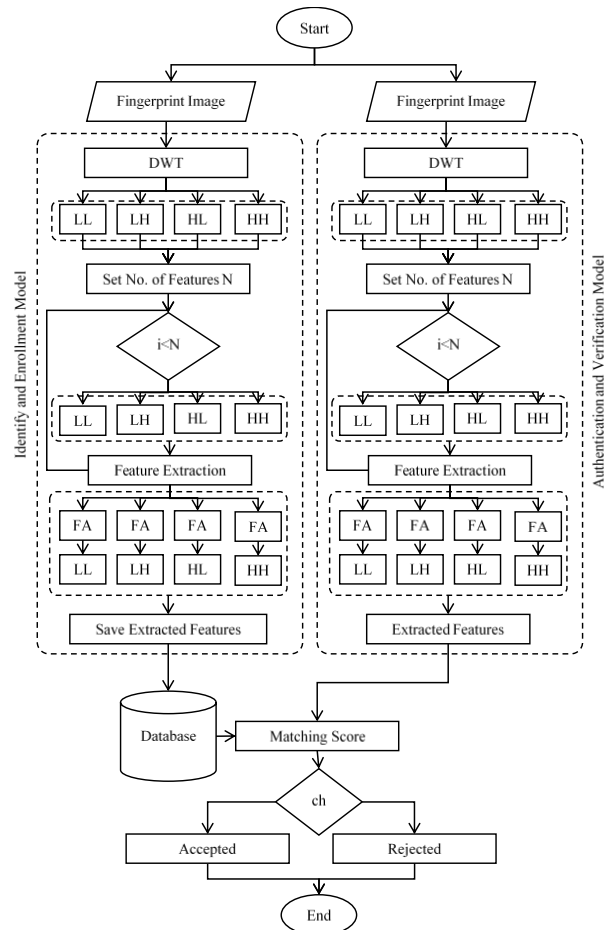


Fig. 3.: Flowchart of Our Proposed System

where  $\alpha_0$  is the initial randomness scaling factor, and  $\delta$  is essentially a cooling factor. For most applications, we can use  $\delta = 0.95$  to  $0.97$  [11]. Regarding the initial  $\alpha_0$ , simulations show that FA will be more efficient if  $\alpha_0$  is associated with the scalings of design variables. Let  $L$  be the average scale of the problem of interest, we can set  $\alpha_0 = 0.01L$  initially. The factor 0.01 comes from the fact that random walks requires several steps to reach the target while balancing the local exploitation without jumping too far in a few steps [9] [14]. The parameter  $\beta$  controls the attractiveness, and parametric studies suggest that  $\beta_0 = 1$  can be used for most applications. However,  $\gamma$  should be also related to the scaling  $L$ . In general, we can set  $\gamma = 1/\sqrt{L}$ . If the scaling variations are not significant, then we can set  $\gamma = 0(1)$ . For most applications, we can use the population size  $n = 15$  to  $100$ , though the best range is  $n = 25$  to  $40$  [11] [12].

#### B. The Efficiency of Firefly Algorithm (FA)

FA is swarm-intelligence-based, so it has the similar advantages that other swarm intelligence-based algorithms have. In fact, a simple analysis of parameters suggests that some PSO variants such as Accelerated PSO [21] are a special case of firefly algorithm when  $\gamma = 0$  [8]. However, FA has two major advantages over other algorithms: automatically subdivision and the ability of dealing with

multimodality. First, FA is based on attraction and attractiveness decreases with distance. This leads to the fact that the whole population can automatically subdivide into subgroups, and each group can swarm around each mode or local optimum. Among all these modes, the best global solution can be found. Second, this subdivision allows the fireflies to be able to find all optima simultaneously if the population size is sufficiently higher than the number of modes. Mathematically,  $1/\sqrt{\gamma}$  controls the average distance of a group of fireflies that can be seen by adjacent groups. Therefore, a whole population can subdivide into subgroups with a given, average distance. In the extreme case when  $\gamma = 0$ , the whole population will not subdivide. This automatic subdivision ability makes it particularly suitable for highly nonlinear, multimodal optimization problems.

In addition, the parameters in FA can be tuned to control the randomness as iterations proceed, so that convergence can also be sped up by tuning these parameters. These above advantages make it flexible to deal with continuous problems, clustering and classifications, and combinatorial optimization as well. As an example, let us use two functions to demonstrate the computational cost saved by FA [9].

**Algorithm (1): Firefly Algorithm**

```

Input: Initialize parameters
 $\gamma \leftarrow$  maximal number of generation
 $r \leftarrow$  light absorption coefficient (distance -light source)
 $d \leftarrow$  domain space
Output: Initialize population
1. Define the objective function of  $f(x)$ , where  $x = (x_1, \dots, x_d)^T$ 
2. Generate the initial population of fireflies or  $x_{i=1..n}$ 
3. Determine the light intensity of  $I_i$  at  $x_i$  via  $f(x_i)$ 
4. While ( $t < \text{Max Gen}$ )
5.   For  $i = 1$  to  $n$  (all  $n$  fireflies)
6.     For  $j = 1$  to  $n$  (n fireflies)
7.       if ( $I_j > I_i$ )
8.         Move firefly  $i \rightarrow j$ 
9.       End if
10.      Attractiveness varies distance  $r \rightarrow [-\gamma r^2]$ 
11.      Evaluate new solutions
12.      Update light intensity
13.     End for j
14.   End for i
15. Rank the fireflies and find the current best;
16. End while
17. Post process results and Visualization
18. End
    
```

C. FA Search Strategy

FA intermittent search strategies concern an iterative strategy consisting of a slow phase and a fast phase [22] [23]. Here the slow phase is the detection phase by slowing down and intensive, static local search techniques, while

the fast phase is the search without detection and can be considered as an exploration technique. For example, the static target detection with a small region of radius  $a$  in a much larger region  $b$  where  $a \ll b$  can be modelled as a slow diffusive process in terms of random walks with a diffusion coefficient  $D$ . Let  $\tau_a$  and  $\tau_b$  be the mean times spent in intensive detection stage and the time spent in the exploration stage, respectively, in the  $2D$  case. The diffusive search process is governed by the mean first-passage time satisfying the following Eq. (4) and (5): [22].

$$D \nabla_r^2 t_1 + \frac{1}{2\pi\tau_a} \int_0^{2\pi} [t_2(r) - t_1(r)] d\theta + 1 = 0 \quad (4)$$

$$u \cdot \nabla_r t_2(r) - \frac{1}{\tau_b} [t_2(r) - t_1(r)] + 1 = 0 \quad (5)$$

where  $t_2$  and  $t_1$  are mean first-passage times during the search process, starting from slow and fast stages, respectively, and  $u$  is the mean search speed. After some lengthy mathematical analysis, the optimal balance of these two stages can be estimated as given in Eq. (6).

$$r_{\text{optimal}} = \frac{\tau_a}{\tau_a^2} \approx \frac{D}{a^2} \frac{1}{\left[2 - \frac{1}{\ln(b/a)}\right]^2} \quad (6)$$

If the search steps have a uniform velocity  $u$  at each step on average, the minimum times required for each phase can be

$$\tau_a^{\text{min}} \approx \frac{D}{2u^2} \frac{\ln^2\left(\frac{b}{a}\right)}{\left[2 \ln\left(\frac{b}{a}\right) - 1\right]} \quad (7)$$

$$\tau_b^{\text{min}} \approx \frac{a}{u} \sqrt{\ln\left(\frac{b}{a}\right) - \frac{1}{2}} \quad (8)$$

estimated as it given in Eq. (7) and (8).

When  $u \rightarrow \infty$ , these relationships lead to the above optimal ratio of two stages

Based on that, the basic steps of the firefly algorithm (FA) can be summarized as the pseudo code shown in Algorithm (1) [23].

D. Landscape-Based Optimality or Firefly Optimized Algorithm (FOA)

The main components of any metaheuristic algorithms are: intensification and diversification, or exploitation and exploration [24] [25]. Diversification means to generate diverse solutions to explore the search space on the global scale, while intensification means to focus on the search in a local region by exploiting the information that a current good solution is found in this region. This is in combination with the selection of the best solutions. Exploration in metaheuristics can be achieved often using randomization [24] [8] [9], which enables an algorithm to can jump out of any local optimum to explore the search globally. Randomization can also be used for local search around the current best if steps are limited to a local region. When the steps are large, randomization can explore the search space on a global scale. Fine-tuning the right amount of randomness and balancing local search and

global search are crucially important in controlling the performance of any metaheuristic algorithm.

Exploitation is the use of local knowledge of the search and solutions found so far so that new search moves can concentrate on the local regions or neighborhood where the optimality may be close; however, this local optimum may not be the global optimality. Exploitation tends to use strong local information such as gradients, the shape of the mode such as convexity, and the history of the search process. A classic technique is the so-called hill-climbing which uses the local gradients or derivatives intensively. In our proposal, and in term to enhance the exploitation of the (FA) we use two additional methods to optimize the solution using global maximum but there is a chance that the solution will get trapped in a local area. A Modified Firefly optimized.

#### E. Landscape-Based Optimality or Firefly Optimized Algorithm (FOA)

The main components of any metaheuristic algorithms are: intensification and diversification, or exploitation and exploration [24] [25]. Diversification means to generate diverse solutions to explore the search space on the global scale, while intensification means to focus on the search in a local region by exploiting the information that a current good solution is found in this region. This is in combination with the selection of the best solutions. Exploration in metaheuristics can be achieved often using randomization [24] [8] [9], which enables an algorithm to can jump out of any local optimum to explore the search globally. Randomization can also be used for local search around the current best if steps are limited to a local region. When the steps are large, randomization can explore the search space on a global scale. Fine-tuning the right amount of randomness and balancing local search and global search are crucially important in controlling the performance of any metaheuristic algorithm.

Exploitation is the use of local knowledge of the search and solutions found so far so that new search moves can concentrate on the local regions or neighborhood where the optimality may be close; however, this local optimum may not be the global optimality. Exploitation tends to use strong local information such as gradients, the shape of the mode such as convexity, and the history of the search process. A classic technique is the so-called hill-climbing which uses the local gradients or derivatives intensively. In our proposal, and in term to enhance the exploitation of the (FA) we use two additional methods to optimize the solution using global maximum but there is a chance that the solution will get trapped in a local area. A Modified Firefly optimized Algorithm (FOA) [8] [9] works better. There is difficulty in selecting probable initial position. So, there are some limitations which affect the accuracy of the FA. The first one is initial position, and the second limitation is the searching redundancy. To make the FOA feature extraction process more efficient and accurate we propose the following improvements to the original FA algorithm

#### 1) Centralization

In this step, we propose to let the FOA start from the center of the fingerprint core and delta extraction. The singularity points are detected using the Poincarè indexes [26] which the index values are equal to  $180^\circ$ ,  $-180^\circ$ ,  $360^\circ$  correspond to a core, a delta and a double core point, respectively. Starting from singularity points, the ROI's extracted as the neighborhood region. This method is based on the Poincarè index since it assumes that the loop, whorl and delta make a Poincarè index equal to  $180^\circ$ ,  $360^\circ$  and  $-180^\circ$ , respectively. Fingerprint singularity regions extraction is composed of three main blocks: directional image extraction, Poincarè indexes calculation and singularity points detection. The fingerprint has many local maximum, yet the original FA depends on selecting just one local maximum and evaluate it with the rest of the features. To avoid misclassification due to mismatched initial point of FOA, we propose to perform this centralization step. Note that from our observation there is still an issue of center synchronization between the fingerprint image and the candidates, so we propose to address this center synchronization problem by using an invariance moment algorithm which can take care of the difference between the extracted fingerprint features and the candidate features. The core point detection that we applied is shown in Fig. 4.b.

#### 2) Discrete Wavelet Transform (DWT)

In this step wavelet transform [27] is used to get an efficient image representation that characterizes the significant image features in compact form. We proposed to use the DWT to apply multi swarm features selection in the same fingerprint image with different direction as a Fig.4.c shows. Two-dimensional discrete wavelet transform (2-D DWT) decomposes a gray-level fingerprint image into one average component sub-band and three detail component sub-bands. The first sub-band is denoted as LL and contains the average components, the second sub-band is denoted as LH and contains vertical edges, and the third sub-band is denoted as HL and contains horizontal edges, and the fourth HH sub-band contains diagonal edges. Features Extraction using FOA is then applied. In this step, the multiple FOA technique is used for extracting the features from the Robust features selection:

In this step, we propose a feature separation concept which relies on using the DWT as a transform to make the PSO move away from the center of the fingerprint to solve the localization problem. Because of the centralization step we proposed, we have a potential problem in the feature selection model: note that the FOA may lose some features because of the rotation and noise. So instead of making the FOA to rely on one region to extract the features, we propose to use the DWT to make the FOA move away from the center by selecting different features using one particle swarm in each sub-band so in total we use four particle swarms in four sub-bands in each fingerprint image. Fig.4.c shows the fingerprint robust feature extraction with the proposed feature separation concept.

**Algorithm (2):** Invariance moment matching algorithm

```

Input: fingerprint candidate features x[i].
Output: Return matching accuracy by comparison the
two set of features between the stored fingerprint
image and the extracted one.
1.   For each data in database the set of stored
features is y[i]
2.   For i=0... N (Number of features) do
3.   |    $\alpha = x[i] * 0.5$ 
4.   |   lower_range= x [i] -  $\alpha$ 
5.   |   high_range= x[i] +  $\alpha$ 
6.   |   If range low <= y[i] and rang high >= y[i]
then |
7.   |   no_check = no_check + 1
8.   |   Else x[i] = 0, y[i] = 0.
9.   |   calculate deference ratio by ratio =
(1/no_check)×100
10.  |   For i=0... N do
11.  |   |   If x[i] <= y[i] then
12.  |   |   error = error + ((x[i]/y[i])×ratio
13.  |   |   Else error = error + ((y[i]/x[i])×ratio
14.  |   Return matching accuracy.

```

3) Robust features selection:

In this step, we propose a feature separation concept which relies on using the DWT as a transform to make the FOA move away from the center of the fingerprint to solve the localization problem. Because of the centralization step we proposed, we have a potential problem in the feature selection model: note that the PSO may lose some features because of the rotation and noise. So instead of making the FOA to rely on one region to extract the features, we propose to use the DWT to make the FOA move away from the center by selecting different features using one particle swarm in each sub-band so in total we use four particle swarms in four sub-bands in each fingerprint image. Fig.4.c shows the fingerprint robust feature extraction with the proposed feature separation concept.

4) Redundancy Avoidance:

According to the FOA steps that shown in algorithm (1), the FOA is going to select the same feature point after some iteration, and that affects the performance of the FOA in speed, so we make sure the FAO doesn't select the same feature that it selected before based on its position x and y. Robust features selection: In this step, we propose a feature separation concept which relies on using the DWT as a transform to make the FOA move away from the center of the fingerprint to solve the localization problem. Because of the centralization step we proposed, we have a potential problem in the feature selection model: note that the FOA may lose some features because of the rotation and noise. So instead of making the FOA to rely on one region to extract the features, we propose to use the DWT to make the FOA move away from the center by selecting different features using one particle swarm in each sub-band so in total we use four particle swarms in four sub-bands in each fingerprint image. Fig.4.d shows the fingerprint robust

feature extraction with the proposed feature separation concept.

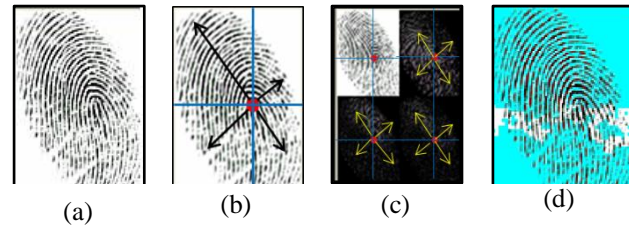


Fig. 4.: A DWT for the fingerprint image with the FOA separation, (a): original fingerprint, (b):Core-point detection, (c): DWT image with centralization (c): different search direction of the FOA for the LL subbands.

F. Invariant Moment Matching

Moment invariants have been frequently used as features for image processing, remote sensing, shape recognition and classification. Moments can provide characteristics of an object that uniquely represent its shape. Invariant shape recognition is performed by classification in the multidimensional moment invariant feature space. Several techniques have been developed that derive invariant features from moments for object recognition and representation. It was Wai [28] that first set out the mathematical foundation for two-dimensional moment invariants and demonstrated their applications to shape recognition. Since the FOA is going to collect the features from the same area which means the set of features are related with each other according to the fitness function of the FOA and comparing with other approach which is depends on select the Minutia features from deferent part inside the image and they are unrelated, FOA features is more accurate for the sequence matching than the other approaches because all the features are related with each other and any effeteness like noise and rotation, so any effeteness will effect just of specific some feature not at all, the neighbor of the current effected one and the fitness function are able to reselect the sext optimal one and the. For this reason, we proposed the invariance moment algorithm as a sequence matching approach for two set of features with the same number that the FOA is going to detect before to test each two set of features alone, first according to the distance between two features, if the candidate features are closer to the extracted that the invariance moment is going to admit it as a true one, but if not the invariance is going to omit it. Second, according to their fitness value this means the real features value. Algorithm (2) shows the steps of the invariance moment matching algorithm [28].

The proposed system relies on the matching operation for granting the recognition. The matching is to match the target person with all people that have enrolled in the database. When the matching (between the database LL features and the extracted LL features) ratio is lower than a first threshold value the person is unauthorized. When it is equal or greater than 98%, the person is authorized. When it is larger than the first threshold, but lower than 98%, then

another comparison between (databases LH, HL, HH features and candidate LH, HL, HH features) is done. If the result of the second comparison is equal or greater than 90% then the person is authorized otherwise the person is unauthorized.

### V. EXPERIMENTAL RESULTS

We test and compare the performance of the proposed system to previous works. To assess the validation and accuracy of the experimental results, a confusion matrix for the detection system is defined as an  $m \times m$  matrix, where  $m$  denotes the number of classes. A confusion matrix contains information about actual and predicted classifications done by a detection system. Performance of such systems is commonly evaluated using the data in the matrix. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The confusion matrix shows the classes which are correctly classified and the classes that are misclassified.

#### A. Databases

The proposed method was evaluated on fingerprint images taken from the public database FVC2002 set\_A [29]. In this database, the fingerprint impressions are acquired by using the modern capacitive and optical sensors, as shown in Table 1.

TABLE 1. SUMMARY OF FOUR FVC2002 DATABASES [29]

	Sensor type	Image size (pixel)	Set A (w×d)	Resolution (dpi)
DB1	Optical sensor	388×374 (142 K)	100×8	500
DB2	Optical sensor	296×560 (162 K)	100×8	569
DB3	Capacitive sensor	300×300 (88 K)	100×8	500
DB3	SFinGe v2.51	288×384 (108 K)	100×8	About 500

FVC2002 contains four distinct subsets: DB1, DB2, DB3, and DB4. Each dataset consists of fingerprint impressions for 100 subjects with eight impressions per subject at various rotations in the range of  $-30^\circ$  to  $+30^\circ$  (determined empirically) and translations. The resolution of the images in FVC2002 is 500 dpi (dots per inch) and in DB2 they are 569 dpi. The second database was generated by selecting the first impression for each subject from all four datasets of FVC2002 by rotating them 10 times by an angle selected randomly between  $-30^\circ$  to  $30^\circ$ . This generated 10 test images for each impression. This database, RFVC2002, was used to demonstrate the robustness of the proposed method for rotation and translation. The third database was collected from 60 individuals using SecuGen Hamster IV FIPS 201/PIV, which is an FBI compliant optical fingerprint scanner. For each finger, 10 images were taken at different rotations, which resulted in a total of 600 images. All the images were scanned at a resolution of 500

dpi as recommended by the FBI and named 'AITDB'. Here, the database capturing was done in a controlled environment to have a resemblance to a real scenario. The empirically observed rotation was in the range of  $[-35^\circ$  to  $40^\circ]$ .

#### B. Overall Dataset performance:

The experiments were conducted to evaluate the performance of the proposed method on the database FVC2002 set\_A (which contains four subsets: DB1, DB2, DB3, and DB4), RFVC2002, and AITDB. The proposed method operates in a verification mode where the false acceptance rate (FAR), the false rejection rate (FRR) and genuine acceptance rate (GAR) are evaluated as:

1) FAR: FAR is defined as the ratio between the numbers of TF and total tested samples as given in (Eq. (9)).

$$FAR = \frac{TP}{TP + TN} \quad (9)$$

2) FRR: FRR is defined as the ratio between the FN and the total tested samples as given in (Eq. (10)).

$$FRR = \frac{TN}{TN + FN} \quad (10)$$

3) Recognition Rate (CVR): Recognition Rate is defined as the ratio between the numbers of correct recognition decision to the total number of attempts as it is given in (Eq. (11)).

$$GAR = (1 - FAR - FRR) * 100 \quad (11)$$

#### C. Components contributions in the proposed system:

Table 2 shows the performance of our system with the combinations of different components, including feature extraction enhancement, DWT, and invariance moments matching. The DWT and invariance moments matching are shown to provide significant performance improvement depending on FAR (false accept rate), FRR (false reject rate), and CAR (correct verification rate),

TABLE 2. COMPONENTS CONTRIBUTION OF THE PROPOSED SYSTEM

Proposed system Components	FAR %	FRR %	CAR %
Feature Enhancement	0.00	10.21	89.79
Feature Enhancement + DWT	0.00	3.60	96.40
Feature Enhancement +DWT+ invariance moments	0.00	0.01	99.99

#### D. Time complexity:

Table 3 describes the time consumed at each stage of the developed verification system.

TABLE 3. THE MEAN OF THE CONSUMED TIME AT EACH STAGE AND THEIR PERCENTAGES RELATIVE TO THE TOTAL TIME

No	Stages	Minutia	Ratio	FOA	Ratio
1	Rad image	0.01 s	0.03%	0.02 s	0%
2	Convert to gray	0.01 s	0.04%	0.00 s	0%
3	Normali zation	0.26 s	0.92%	0.00 s	0%
4	Filtering	0.31 s	1.09%	0.00 s	0%
5	Binriliza tion	0.05 s	0.19%	0.00 s	0%
6	ROI	0.01 s	0.02%	0.00 s	0%
7	Island removal	2.44 s	9.76%	0.00 s	0%
8	Hole removal	0.88 s	3.09%	0.00 s	0%
9	Thinning	0.08 s	0.27%	0.00 s	0%
10	Edge Linking	0.05 s	0.16%	0.00 s	0%
11	Features Extractio n	0.22 s	0.78%	0.08 s	0%
12	Partitioni ng	23.91 s	83.6%	0.00 s	0%
13	Moment Analysis	0 ms	0%	0.00 s	0%
14	Matchin g	0 ms	0%	0.00 s	0%
Total time		28.58	100%	0.1 s	100%

E. Robustness to rotation

Table 4 shows the difference between the Minutiae feature based approach and the PSO feature based approach in a rotated fingerprint matching.

TABLE 4. THE MATCHING ACCURACY FOR ROTATED FINGERPRINT

No	Sample Rotated	PSO Matching Accuracy	Minutiae Matching Accuracy
1	15°	98.55%	0.00%
2	45°	96.08%	0.00%
3	90°	98.81%	0.00%
4	125°	98.55%	0.00%
5	Flipping 180°	100.00%	0.00%
6	225°	99.58%	0.00%
7	270°	98.36%	0.00%
8	315°	99.71%	0.00%

F. Comparison with Other Matchers-Based Methods

A comparison of our proposed method is made here with other mache-based approaches, which include the

following methods. GABOR using (Gabor filter) is an image-based matcher to describe the fingerprint image [17]. LBP, local binary pattern based descriptor [18]. DCT, discrete cosine transform based fingerprint matcher [19]. WFMT, Fourier-Mellin transform based matcher [20]. We used the same parameters for experiments, as were used in the above methods in the literature. These methods were tested on FVC2002 for the comparative performance analysis in terms of EER (%). For evaluating the similarity between test and trainee fingerprint images, the Euclidian distance was employed. The results of these comparisons are given in Table 5.

TABLE 5. COMPARE PROPOSED METHOD WITH PREVIOUS STUDY ON FCV2002 [29]

Method	DB1	DB2	DB3	DB4	Avera ge
GABOR [17]	1.87	3.98	4.64	6.21	4.17
LBP [18]	7.00	6.2	9.9	7.5	7.65
DCT [19]	2.96	5.42	6.79	7.53	5.68
WFMT [20]	2.43	4.41	5.18	6.62	4.66
Orientation Features [30]	3.12	2.89	4.34	3.4	3.43
Proposed FOA	<b>0.0</b>	<b>1.2</b>	<b>0.1</b>	<b>2.4</b>	<b>1.23</b>

VI. CONCLUSION

In this paper, we propose a FOA based fingerprint authentication system that provides several important functionalities. First, unlike Minutiae features, the FOA features are invariant to rotation, and during the searching process, FOA is going to determine the ROI (region of interest) whereas the other algorithms spent some time to do that. Second, the proposed FOA will directly detect the core point. Third the proposed FOA will collect multi features in different directions by using DWT which solves the localization problem. Finally, the proposed FOA is shown to provide excellent performance, with 100 % accuracy in a frontal situation and between 96 to 99.5% in a rotated situation.

REFERENCES

- [1] S.J. Xie, J. Yang, D.S. Park, S.Yoon, and J. Shin (2011), "Fingerprint Quality Analysis and Estimation Approach for Fingerprint Matching"; In: J. Yang, and L. Nanni (Eds.), State of the art in Biometrics, In Tech, 2011.
- [2] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar, "Handbook of Fingerprint Recognition", First Edition, Springer, New York, 2003.
- [3] A. Rawat, "A Hierarchical Fingerprint Matching system", M.Sc Thesis, Indian Institute of Technology Kanpur, July 2009.
- [4] A. Ross, A. Jain, and J. Reisman, "A Hybrid Fingerprint Matcher", Pattern Recognition, ISSN: 0031-3203, Vol. 36, No. 7, pp. 1661–1673, 2003.
- [5] A.N. Ouzounoglou, P.A. Asvestas and G.K. Matsopoulos, "A New Approach in Fingerprint Matching based on a Competitive Learning Algorithm", International Journal of Information and Communication Technology Research, Vol. 2, No.9, pp. 723-731, 2013.
- [6] Huong Thuy, Nguyen Thi," An efficient method for fingerprint matching based on local point model", Computing, Management



- and Telecommunications (ComManTel), International Conference, 2013.
- [7] J. Kennedy and R. Eberhart, Particle swarm optimisation, in: Proc. of the IEEE Int. Conf. on Neural Networks, Piscataway, NJ, pp. 1942-1948 (1995).
- [8] X. S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, UK, (2008).
- [9] X. S. Yang, Firefly algorithms for multimodal optimisation, Proc. 5th Symposium on Stochastic Algorithms, Foundations and Applications, (Eds. O. Watanabe and T. Zeugmann), Lecture Notes in Computer Science, 5792: 169-178 (2009).
- [10] X. S. Yang and S. Deb, Cuckoo search via Lévy flights, Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009, India), IEEE Publications, USA, pp. 210-214 (2009).
- [11] J. C. Bansal and K. Deep, Optimisation of directional overcurrent relay times by particle swarm optimisation, in: Swarm Intelligence Symposium (SIS 2008), IEEE Publication, pp. 1-7, (2008).
- [12] C. A. Floudas and P. M. Pardalos, Encyclopedia of Optimisation, 2nd Edition, Springer (2009).
- [13] R. S. Parpinelli and H. S. Lopes, New inspirations in swarm intelligence: a survey, Int. J. Bio-Inspired Computation, 3, 1-16 (2011).
- [14] X. S. Yang, Engineering Optimisation: An Introduction with Metaheuristic Applications, John Wiley and Sons, USA (2010).
- [15] X. S. Yang, A new metaheuristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimisation (NICSO 2010) (Eds. J. R. Gonzalez et al.), Springer, SCI Vol. 284, 65-74 (2010).
- [16] A. H. Gandomi, X. S. Yang, and A. H. Alavi, Cuckoo search algorithm: a metaheuristic approach to solve structural optimisation problems, Engineering with Computers, 27, article DOI 10.1007/s00366-011-0241-y, (2011).
- [17] A. Ross, A. K. Jain, and J. Reisman, "A hybrid fingerprint matcher," Pattern Recognition, vol. 36, no. 7, pp. 1661-1673, 2003.
- [18] M. Tico, E. Immonen, P. Ramo, P. Kuosmanen, and J. Saarinen, "Fingerprint recognition using wavelet features," in Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2001), Sydney, Australia, 2001, pp. 21-24.
- [19] T. Amornraksa and S. Tachaphetpiboon, "Fingerprint recognition using DCT features," Electronics Letters, vol. 42, no. 9, pp. 522-523, 2006.
- [20] A. T. B. Jin, D. N. C. Ling, and O. T. Song, "An efficient fingerprint verification system using integrated wavelet and Fourier-Mellin invariant transform," Image and Vision Computing, vol. 22, no. 6, pp. 503-513, 2004.
- [21] X. S. Yang, S. Deb and S. Fong, (2011). Accelerated particle swarm optimization and support vector machine for business optimization and applications, Networked Digital Technologies (NDT'2011), Communications in Computer and Information Science, Vol. 136, Part I, pp. 53-66.
- [22] O. B'énichou, C. Loverdo, M. Moreau and R. Voituriez, Two-dimensional intermittent search processes: An alternative to Lévy flight strategies, Phys. Rev., E74, 020102(R), (2006).
- [23] O. B'énichou, C. Loverdo, M. Moreau, and R. Voituriez, Intermittent search strategies, Review of Modern Physics, 83, 81-129 (2011).
- [24] C. Blum and A. Roli, Metaheuristics in combinatorial optimisation: Overview and conceptual comparison, ACM Comput. Surv., Vol. 35, 268-308 (2003).
- [25] X. S. Yang, Swarm-based metaheuristic algorithms and no-free-lunch theorems, in: Theory and New Applications of Swarm Intelligence (Eds. R. Parpinelli and H. S. Lopes), Intech Open Science, pp. 1-16 (2012).
- [26] H.Zhang, Y.Yin and G.Ren, "An Improved Method for Singularity Detection of Fingerprint Images", Book Advances in Biometric Person Authentication, Springer Berlin/Heidelberg, Vol. 3338/2004, pp.516-524, ISBN 978-3-540-24029-7.
- [27] Adil Abdulwahhab Ghidan Al-Azzawi, Muneera abed hmidi Al-Saedi and Asmaa A, "Speeding and Efficiency Increasing of Color Biometric Finger Print Identification Using Particle Swarm Optimization Based on Selected Feature by Multiwavelet", Internet survey, <http://www.cms.livjm.ac.uk/pgnet/Proceeding/papers/m1569449223-mehdi.pdf>, 2011.
- [28] Wai-Hong Wong, Wan-Chi Siu, Kin-Man Lam, "Generation of moment invariants and their uses for character recognition", department of Electronic Engineering, Hong Kong Polytechnic, Hung Hon, Kowloon, Hong Kong, February 1995.
- [29] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2002: second fingerprint verification competition," in Proceedings of the 16th International Conference on Pattern Recognition, Quebec City, Canada, 2002, pp. 811-814.
- [30] Ravinder Kumar, Pravin Chandra, and Madasu Hanmandlu, "A Robust Fingerprint Matching System Using Orientation Features", J Inf Process Syst, Vol.12, No.1, pp.83-99, March 2016.