Neural Computing and Applications
https://doi.org/10.1007/s00521-021-06705-0

ORIGINAL ARTICLE

Finger knuckle biometric feature selection based on the FIS_DE optimization algorithm

P. Jayapriya[1] · K. Umamaheswari[1]

Received: 31 October 2020 / Accepted: 27 October 2021

Abstract

In recent years, the hand-based biometric system has received significant attention in identifying a person. In the hand-based biometric, the finger knuckle print plays a vital role in recognizing a person. The main purpose of this research is to propose an effective feature optimization technique for identifying the best feature vectors for finger knuckle print-based authentication. This work presents a novel feature selection algorithm, fitness index selection with differential evolution (FIS_DE), based on K-nearest neighbor (KNN). Initially, the feature extraction is performed using the conventional methods like principal component analysis, linear discriminant analysis, and independent component analysis. Then, evolutionary algorithm is used for feature selection with best vectors. The population-based metaheuristic algorithm DE proposed is used to optimize the KNN classifier. FIS_DE_KNN is compared with Euclidean and neural network classifiers to show the improved efficiency of the proposed work. In this research, experimental results of the proposed FIS_DE-KNN improve the classification accuracy with an optimized number of features.

Keywords Finger knuckle print · Feature extraction · Feature selection · Differential evolution · Classification · K-nearest neighbor

Abbreviations

FKP	Finger knuckle print
BL_POC	Band-limited phase-only correlation
GA	Genetic algorithm
FS	Feature selection
FSGS	Feature selection gaining-sharing knowledge-based
EGA	Exponential genetic algorithm
SCMO	Speed-constrained multi-objective
PSO	Particle swarm optimization
FRR	False rejection rate
FAR	False acceptance rate
PCA	Principal component analysis
LDA	Linear discriminant analysis
KNN	K-nearest neighbor

1 Introduction

A biometric recognition is an effective system that automatically authenticates an individual, based on physiological or behavioral traits. This system is used in applications such as banking security, surveillance, and logical access control [1–3]. Over the past few decades, the usage of biometric features (iris, fingerprint, voice, face, ear, palm print, hand geometry) is exhaustively analyzed with computing systems [4, 5]. Hand-based recognition systems (hand vein, palm print, hand geometry, palm print, FKP) attracted considerable attention among the researchers [6]. Hand-based biometrics are user-friendly, reliable, and low cost for personal authentication [7]. A hand-based recognition system, compared to other systems, performs better in varying lighting conditions. Moreover, hand-based features are unique over time, economical and straightforward data collection capabilities, and suitable for outdoor and indoor usage [8–10]. These features make hand-based recognition a user-friendly technique in the biometric fields [11]. The FKP-based recognition system primarily authenticates an individual's identity by analyzing and measuring the texture pattern and knuckle lines [12, 13].

✉ P. Jayapriya
1707ri02@psgtech.ac.in

K. Umamaheswari
uma.it@psgtech.ac.in

[1] Department of Information Technology, PSG College of Technology, Coimbatore, Tamil Nadu, India

Published online: 04 January 2022

Conflicting classifier scores reduce the efficiency of the majority of existing FKP-based recognition systems in a complex environment [14, 15]. To address these issues, this paper proposes a new feature selection algorithm based on differential evolution. In this work, input images were collected from the PolyU finger knuckle dataset. Then image pre-processing was accomplished using histogram equalization, which delivers an excellent enhancement of image contrast. Then, hybrid feature extraction was performed by various fusions of PCA, ICA, and LDA to extract the features from enhanced finger knuckle images. The hybrid features are fed into the proposed feature selection technique FIS_DE to reduce the dimensionality of the feature vectors.

The multi_instance fusion with normalization scores is a new approach proposed by Shariatmadar and Faezthat [7] which gave improved performance. But due to illumination conditions, research in [7] degraded the recognition accuracy. Finger knuckle print (FKP) recognition based on various features such as discriminative features and texture features achieves better accuracy with a lower error rate [8, 16, 17]. The adaptive decision level fusion combines decision and score level fusion on FKP images. It manages the uncertainty in the decision by decreasing the error in fuzzy sets [18]. The local and global features of FKP images based on the band-limited phase-only correlation (BL_POC) exhibits high authentication performance compared to the traditional systems [19].

Feature selection (FS) is a technique used to select the subset of features from the original feature set of the image. The main objective of feature selection is to recognize a pattern with the smallest possible features while also reducing the time and memory complexity of the original search space. The goal is attained by eliminating the redundant and irrelevant features without losing the essential information of the image. As a consequence, different methodologies have been suggested to deal with the situation. Like, exhaustive search, greedy search, random search, and other methods have been used to identify the best subset of features. The majority of the methods suffer from premature convergence, excessive complexity, and high computational cost [20]. More efficiently the metaheuristic algorithms are implemented to select the relevant features. Some feature selection techniques based on metaheuristic algorithms are discussed in this section.

The Genetic Algorithm (GA) is used for feature selection in biometrics such as ear [21, 22], fingerprint [23], and iris [23]. Particle swarm optimization (PSO) is used in face recognition to choose the feature subset of face images and guarantee significant results as compared to other metaheuristic algorithms such as genetic algorithms and ant colony optimization techniques [24].

Prachi et al. [20] investigated two approaches in feature selection algorithms, including a binary variant of the feature selection gaining-sharing knowledge-based optimization algorithm (FS-BGSK) and a population reduction technique with BGSK, to improve feature selection exploration and exploitation efficiency. The feature selection problem related to metaheuristic algorithms is clearly described in the literature [25]. According to Parham Moradi and Mozhgan Gholampour [26], the hybrid practical swarm optimization employs a local search strategy to select the best feature subset.

Reyhaneh et al. [27] suggested a nonlinear mapping function-based grasshopper optimization algorithm for feature selection. This proposed algorithm preserves exploration and exploitation by enhancing accuracy and computational time.

Palisetty and Gogulamanda [28] have developed the exponential genetic algorithm (EGA) for palmprint recognition. Significant improvements in accuracy and reductions in space and time are demonstrated by varying the fitness function and leveraging the exponential function.

Larbi and Giouzelis [29] presented a face recognition speed-constrained multi-objective particle swarm optimization (SMPSO) for feature selection. Khushaba et al. [30] proposed a differential Evolution (DE) with real-valued optimizer modification to reduce computational cost and improve accuracy for Brain-computer Interface.

For face recognition, Maheswari et al. [31] compare the differential evolution (DE) and genetic algorithm (GA) feature selection methods. Here, features are extracted using the local directional pattern (LDP) and the local binary pattern (LBP). Based on support vector machine classification, DE outperforms well than GA.

The feature Selection is applied in various biometrics such as the face, palmprint, and finger vein. The researchers of the study focus on feature extraction techniques for finger knuckle biometrics, and only some papers illustrate the FS for various unimodal and multimodal FKP biometrics. Differential Evolution is an effective evolutionary technique used in a wide range of applications due to characteristics such as ease of implementation, small control parameters, robustness, and computational efficiency [32]. The proposed fitness and index selection based on differential evaluation (FIS_DE) method for FKP biometric is used to find the best features with the best recognition accuracy.

In this work, FIS_DE represents the data with an optimal subset of features, which alleviates the "curse of dimensionality" concern. The KNN classification methodology was used to classify the individual as an authorized or unauthorized person using the output of feature optimization. The performance of the proposed

work evaluated using accuracy, false acceptance rate (FAR), false rejection rate (FRR), recall, and precision.

The organization of the paper is as follows: Sect. 2 presents the proposed work for feature selection. Section 3 states the quantitative and comparative results of the proposed work. Finally, the concluding remarks are shown in Sect. 4.

2 Proposed fitness index selection with differential evolution system

The paper proposes a new feature optimization technique to improve recognition accuracy. Pre-processing, feature extraction, feature fusion, feature selection, and classification are the five phases of the proposed authentication system. The block diagram of the proposed work is shown in Fig. 1, and the detailed description of the proposed work is given below.

2.1 Dataset description

The input data have been collected from the PolyU finger knuckle dataset [33, 34]. This dataset was collected by the IIT Delhi campus and the Hong Kong Polytechnic University campus using a handheld camera between 2006 and 2013. The PolyU finger knuckle dataset contains 2500 + finger dorsal images segmented from 500 + subjects' middle fingers. In the PolyU finger knuckle dataset, 88% of the subjects are below 30 years which includes both male and female candidates. Figure 2 shows the sample finger dorsal images as well as the middle fingers.

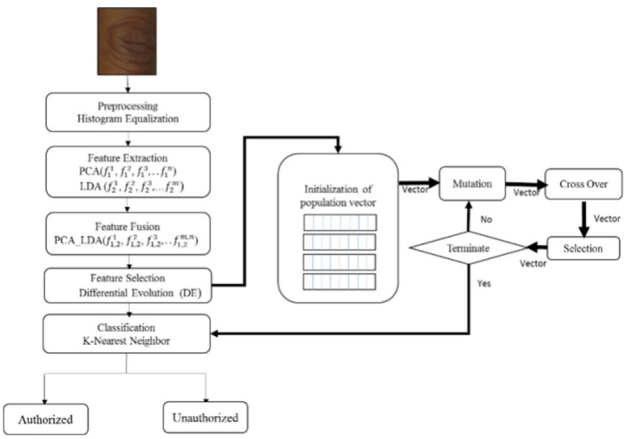

Fig. 1 Workflow of the proposed system

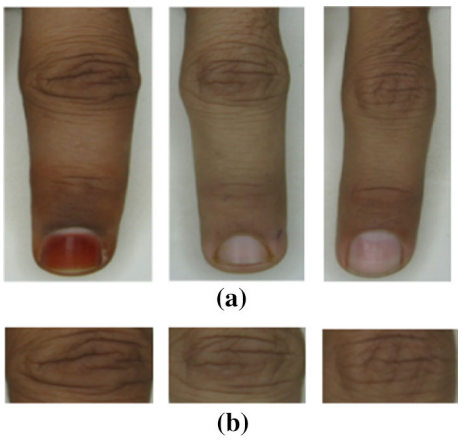

Fig. 2 **a** Middle finger images, **b** segmented finger dorsal images

2.2 Data pre-processing using histogram equalization

The pre-processing is accomplished to reduce the noise and to enhance the quality of the images. In this study, histogram equalization is utilized as a pre-processing approach for finger dorsal images.

I is considered as an input finger knuckle image, which is determined by matrix $M(m_r by m_c)$ that ranges between 0 and $L-1$. The histogram equalized finger knuckle image H is represented in Eq. (1).

$$H_{i,j} = \text{floor}(L-1)\sum_{n=0}^{I(i,j)} P_n \tag{1}$$

where L is indicated as an intensity value that ranges from 0 to 256 and i and j represent the row and column of the matrix of the image.P is denoted as a normalized histogram of input finger knuckle image I,n is stated as the number of images, and floor() is round off the nearest integers equivalent to transforming the pixel intensity K of I which is indicated in Eq. (2).

$$T(K) = \text{floor}((L-1)\sum_{n=0}^{K} P_n) \tag{2}$$

In transformation, the pixel intensities of input finger knuckle image I are considered as the continuous random variables of X, Y on $[0, L-1]$ with Y that is expressed in Eq. (3).

$$Y = T(X) = (L-1)\int_0^x p_X(x)\mathrm{d}x \tag{3}$$

where T is signified to be cumulative distributive function of X multiplied by $(L-1)$ and p_X is indicated as the probability density function of input finger knuckle image I. In this scenario, Y is defined by $T(X)$ that is uniformly distributed over $[0, L-1]$, namely $P_Y(y) = \frac{1}{L-1}$, which is

stated in Eqs. (4), (5), and (6). Figure 3 represents the sample input and pre-processed image.

$$\int_0^y P_Y(z)\mathrm{d}z = \frac{1}{L-1} = \int_0^{T^{-1}(y)} P_X(w)\mathrm{d}w \tag{4}$$

$$\frac{\mathrm{d}}{\mathrm{d}y}\left(\int_0^y P_Y(z)\mathrm{d}z\right) = P_Y(y) = P_X\left(T^{-1}(y)\right)\frac{\mathrm{d}}{\mathrm{d}y}\left(T^{-1}(y)\right) \tag{5}$$

$$\frac{\mathrm{d}T}{\mathrm{d}x}_{|x=T^{-1}(y)}\frac{\mathrm{d}}{\mathrm{d}y}\left(T^{-1}(y)\right) = (L-1)P_X\left(T^{-1}(y)\right)\frac{\mathrm{d}}{\mathrm{d}y}\left(T^{-1}(y)\right) = 1 \tag{6}$$

2.3 Hybrid feature extraction

The feature extraction is carried out for extracting the feature vectors from the denoised images. In this proposed work, feature extraction is done by utilizing PCA, ICA, and LDA from the denoised FKP images. Detailed explanations about the feature extraction methodologies are presented below.

2.3.1 Principal component analysis

Usually, PCA is a statistical scheme that is used for extracting the feature vectors from the pre-processed FKP images. PCA delivers more attention on variance and covariance of the new values $x_1, x_2, x_3, ..x_p$. These magnitude values are higher compared to other values because the obtained magnitude values attain heavyweights. To address this concern, the magnitude values are determined on scales with different ranges, or else the measurement units are not equal then R denote a correlation matrix, which is estimated from n observations on every principal component p of random values. Eigenvector and eigenvalue pairs of R are stated as (ε_1, e_1), $(\varepsilon_2, e_2), (\varepsilon_3, e_3), \ldots ...(\varepsilon_p, e_p)$. The i^{th} sample principal component of a vector $x = x_1, ...x_p$ is denoted in Eq. (7).

(a) (b)

Fig. 3 **a** Original image, **b** pre-processed image

$$e_i Z = e_{i1}Z_1 + e_{i2}Z_2 + \cdots e_{ip}Z_p, \quad i = 1, 2, 3, ..p \tag{7}$$

where $e_i Z = (e_{i1}, e_{i2}, e_{i3}, ...e_{ip})$ is indicated as ith eigenvalue and $Z = Z_1, Z_2, Z_3, \ldots .Z_p$ is denoted as a standardized observation of vectors. In PCA, the covariance pairs are zero, and the sample variances are illustrated as n_i. Additionally, the variance in all standardized variables is equal to the variance in the principal components. The standardized observation of vector is expressed in Eq. (8).

$$Z_k = \frac{x_{k-\overset{\cdot}{x}_k}}{\sqrt{v_{kk}}}, \quad k = 1, 2, 3, \ldots p \tag{8}$$

where x_k is denoted as mean and v_{kk} is stated as the variable variance of x_k.

2.3.2 Independent component analysis

ICA is a linear approach that separates additive sub-components from the multi-variate components. Hence, the separated parts are non-Gaussian that is free from the existing independent schemes. Therefore, ICA effectively identifies the independent source points from the linear mixture components and helps to solve the issues related to higher-order data. By employing image decomposition and representation, ICA delivers a significant data representation. The primary data model utilized in ICA is indicated as $x(t) = [x_1(t), ..x_n(t)]^T$. Vectors $x(t)$ have a standard unknown zero mean non-Gaussian statistical distribution, which is shown in Eq. (9).

$$x(t) = As(t) = \sum_{j=q}^{m} s_j(t)a_j \tag{9}$$

where the vector $s(t) = [s_1(t), \ldots .s_m(t)]^T$ comprises m independent components $s_j(t)$ for the data vectors $x(t)$, and $A = [a_1, \ldots .a_m]$ is denoted as constant total rank $n \times m$ matrix named as mixing matrix. In Eq. (9), the number of independent components m is equal to the number of mixtures n. The independent parts are identified by $m \times n$ inverse mapping as stated in Eq. (10).

$$y(t) = Bx(t) \tag{10}$$

Thus, the m vector becomes an estimate $y(t) = \hat{s}(t)$ of the independent component vector $s(t)$. The estimate of mixing matrix A is attained from the pseudo inverse of B. In the ICA approach; the basis vector a_j is not orthogonal.

2.3.3 Linear discriminant analysis

LDA is an angler's linear discriminant that is utilized to identify the linear combinations of features. Generally, LDA finds the directions to achieve maximum class discrimination. To perform this action, within- and between-

class scatter matrices are determined. If the number of classes is K, then the within-class scatter matrix Σ_w is the scatter of samples around the type means $m_i, i = 1, \ldots .K$ that is represented in Eqs. (11) and (12)

$$\Sigma_w = \sum_{i=1}^{K} p(w_i) E[(x - m_i)(x - m_i)^t, x \epsilon w_i] \tag{11}$$

$$\Sigma_w = \sum_{i=1}^{K} p(w_i) \Sigma_i \tag{12}$$

where Σ_i and $p(w_i)$ are represented as covariance matrix and probability of the ith class w_i. Respectively, the between-class scatter matrix Σ_B is the scatter of class mean $m_i, i = 1, \ldots .K$ around the mixture mean m, which is denoted in Eq. (13).

$$\Sigma_B = \sum_{i=1}^{K} p(w_i)(m - m_i)(m - m_i)^t \tag{13}$$

Hence, the mixture scatters matrix is the covariance of all samples, irrespective of the class assignments as shown in Eq. (14). All features belong to a similar class are combined, and the different features are distinct from attaining high-class separability.

$$\Sigma = \Sigma_B + \Sigma_w \tag{14}$$

2.3.4 Feature level fusion

The three subspace algorithms, such as PCA, LDA, and ICA, are used to extract the features for the index finger knuckle. Consider $F1 = \{f_1^1, f_1^2, \ldots . f_1^m\}, F2 = \{f_2^1, f_2^2, \ldots ., f_2^n\} and F3 = \{f_3^1, f_3^2, \ldots ., f_3^p\}$ are the three feature vectors obtained from the FKP. These feature vectors are concatenated to build a new feature vector (W) with various possible combinations ($F1 + F2 + F3$, $F1 + F2$, $F1 + F3$, $F2 + F3$). The result of the fusion vector is denoted by

- $W_1 = \{f_1^1, f_1^2, \ldots . f_1^m, f_2^1, f_2^2, \ldots ., f_2^n f_3^1, f_3^2, \ldots ., f_3^p\}$,
- $W_2 = \{f_1^1, f_1^2, \ldots . f_1^m, f_2^1, f_2^2, \ldots ., f_2^n\}$,
- $W_3 = \{f_1^1, f_1^2, \ldots . f_1^m, f_3^1, f_3^2, \ldots ., f_3^p\}$,
- $W_4 = \{f_2^2, \ldots ., f_2^n f_3^1, f_3^2, \ldots ., f_3^p\}$.

The pseudo code of the of the feature level fusion is described in (a).

a) Proposed multi-algorithm feature level fusion

The feature vectors obtained from the multi-algorithm are classified by Euclidean distance, K-nearest neighbor (KNN), and neural network (NN). The KNN classification accuracy provides an improved result. The pseudo-code for the proposed feature level fusion is depicted in Algorithm 1.

Algorithm 1. Proposed Feature Level Fusion

For each FKP image, do
I_FKP = Preprocess_histogram_Equalization (FKP_Image);
Feature Vector (FV) = Feature_extraction {PCA (I_FKP); LDA (I_FKP); ICA (I_FKP)};
Fusion_FV = concatenation (Feature Vector (PCA (I_FKP); LDA (I_FKP); ICA (I_FKP)) ;
end for

result_Euclidean = Euclidean distance (Fusion_FV);
result_KNN = result_KNN _classification (Fusion_FV);
result_NN = result_NN _classification (Fusion_FV);
Compare (result_Euclidean, result_KNN, result_NN);
Best = result_KNN

2.4 Feature selection based on FIS_differential evolution

The differential evaluation with fitness and index-based selection is described in detail. Here the motivation and FIS are explained, and the FIS_DE algorithm is presented.

2.5 FIS

A new selection method is introduced to use the outcome of best vectors based on fitness. The new simple proposed method is called fitness and index-based selection (FIS).

The fitness is based on the individual recognition accuracy and fitness accuracy (F_A) for $x_{n,i}^g$ where g is the generation and n is a population, i = 1, 2, 3 … D. For each current population, the max accuracy is calculated and is defined as in Eq. (15)

$$F_A\left(x_{n,i}^g\right) = \text{Max Accuracy}_{\text{best}}\left(x_{n,1}^g, x_{n,2}^g, x_{n,3}^g, \ldots x_{n,D}^g\right) \tag{15}$$

The $\text{MaxAccuracy}_{\text{best}}$ is the best index vector of an individual in the current population in generation (G). The FIS is calculated based on the fitness accuracy of the number of generations.

FIS_DE defines the best vector of an individual for each target vector. The FIS_DE utilizes the image information for optimizing the features which satisfy the objective function. The FIS_DE improves the sense of balance between exploitation and exploration. The empirical analysis of FIS_DE is shown in Sect. 3 with two different image sizes represented as dataset A and dataset B.

2.6 FIS_D E

Feature selection entails locating the most important features that enable a specific objective function to be optimized.

Objective function: To evaluate the recognition rate of the biometrics in terms of accuracy.

In this proposed work, fitness of is based on the accuracy and is calculated using KNN classifier and the formula is given in Eq. (16)

$$\text{Obj} = \text{Max}\ (\text{Accuracy}) \tag{16}$$

After feature extraction, optimization is carried out by utilizing the FIS_differential evolution (DE) optimization algorithm. DE is a population-based stochastic model which performs well in global optimization. DE reduces the complexity of the computational operation and performance robustness and good convergence results without using optimum resources. The performance of the DE depends upon the strategy and the control parameters. The basic procedure performed in the algorithm is crossover, mutation, and selection. The control parameters are the following: NP population size, F scaling factor, and Cr Crossover rate.

Since the DE is population based, the initial population is randomly initialized. The X is a vector that should be optimized. Let X_i^* where i = 1, 2, 3 … D, $X \in R^D$ with boundary conditions of $L \leq X \leq H$. x is an optimal solution with D-dimensional space. The population matrix is defined in Eq. (17)

$$x_{n,i}^g = \left[x_{n,1}^g, x_{n,2}^g, x_{n,3}^g, \ldots x_{n,D}^g\right] \tag{17}$$

where g is the generation, N is population size, and n = 1, 2, 3 … N.

The initial population is randomly generated between lower bound (L) and upper bound (H) boundary constraints followed by

$$x_{n,i} = x_{n,i}^L + \text{rand}() * \left(x_{n,i}^H - x_{n,i}^L\right) \quad i = 1, 2, 3, \ldots D \text{ and } n = 1, 2, 3, \ldots N \tag{18}$$

where $x_i^L and x_i^H$ are the lower bound and upper bound of the variable x_i. After initialization, the evaluation starts with mutation, crossover, and selection. DE starts the search process with the *direction* and *distance* based on the information from the current population. The three vectors $x_{r1n}^g, x_{r2n}^g, and x_{r3n}^g$, are selected randomly from each parameter vector $n = 1, 2, 3 \ldots N$ where $x_{r1n}^g \neq x_{r2n2}^g \neq x_{r3n}^g$.

The mutation is the main distinctive factor in DE. The trial vector is obtained from the target and difference vector. Various mutation strategies are there to generate the mutant vector. The mutant vector is generated by the classic mutation strategy by Eq. (19). The classic mutation is DE/rand/1/bin, where DE is differential evolution, *rand* indicates that the vectors are randomly selected, "1" represents the number of different pairs, here it is one pair, and the bin represents the binomial distribution. The mutant vector is generated using Eq. (19)

$$v_n^{g+1} = x_{r1n}^g + F\left(x_{r2n}^g - x_{r3n}^g\right) \tag{19}$$

where n = 1, 2, 3, … N and $r1$, $r2$, $r3$ are the random vectors dissimilar from each other, and F is the scaling factor range between 0 and 1.

The next step is to complete the mutation strategy of the DE process with a uniform crossover which constructs the trial vector. The selected random vector from the population is also known as target vector $x_{n,i}^{g+1}$, which is a crossover of the mutant vector $v_{n,i}^{g+1}$ to derive a trial vector $v_{n,i}^{g+1}$.

$$u_{n,i}^{g+1} = \begin{cases} v_{n,i}^{g+1} & \text{if rand}() \leq \text{CR or } i = I_{\text{rand}} \quad i = 1, 2.3, \ldots D \text{ and} \\ x_{n,i}^g & \text{if rand}() > \text{CR and} i \neq I_{\text{rand}} \quad n = 1, 2, 3, \ldots N \end{cases} \tag{20}$$

I_{rand} is a random number between $[1, D]$, CR is the cross - over propability range between $[0, 1]$.

In selection, the vector with the lowest value is selected between the target vectors $x_{n,i}^g and$ the trial vector $u_{n,i}^{g+1}$ for the next generation. The proposed FIS_DE is illustrated in Algorithm 2.

$$x_n^{g+1} = \begin{cases} u_n^{g+1} & \text{if } f\left(u_n^{g+1}\right) < f\left(x_n^g\right) \\ x_n^g & \text{Otherwise} \end{cases} \tag{21}$$

Algorithm 2. FIS_Differential Evolution

Step 1: Initialize the population randomly of size NP
Step2: Calculate the fitness values of all individuals in the population
Step 3: Repeat
Step 4: Generate mutant vector using mutant strategy in eq (20)
Step 5: Generate the trial vector using crossover in eq (21)
Step 6: Select the lowest value vector between target and trial vector for the next generation eq (22)

Step 7: Select the impact of each individual based on the best fitness eq (15)
Step 8: Compute the selection probability of each individual's index vector.
Step 9: Select a unique index vector based on probabilities.
Step 10: Until (termination condition reaches maximum number of generation

2.6.1 Complexity analysis of FIS_DE

The aim of this proposed FIS_DE is to eliminate redundant and unwanted features in order to select optimal features. The complexity of the traditional DE is O (G max x NP x D), and G is the number of generations. In the training phase, the FIS_DE is used to train the images.

Approximately 30 runs, 100 iterations for each run, and the influence of an individual index vector are represented with a reduced number of features. Each image takes 2.3 s to train for random initialization of index vector 10, while random initialization of index vector 5 takes 1.3 s for a single image. The query image's index vector is directly used as input in the testing phase to compare with the trained image using KNN classification. Since the selected index-based feature vector of the query image is only used to identify a user. The complexity of FIS_DE reduces the user's computational complexity and storage memory requirements.

a) Proposed reduced feature selection algorithm

In the previous algorithm, the features extracted directly from the algorithm are used as input to the classification, which consumes a lot of memory and slows down the recognition speed. The redundant and noise features are removed from the concatenated feature vector (Fusion FV), reducing the feature vector size. From the fusion FV, the differential evolutionary (DE) algorithm selects the best attributes.

In Algorithm 3, the proposed feature selection from the concatenated feature a vector is shown.

Algorithm 3. Proposed Reduced Feature Selection

For each FKP image, do
I_FKP = Preprocess_histogram_Equalization (FKP_Image);
Feature Vector (FV) = Feature_extraction {PCA (I_FKP); LDA (I_FKP); ICA (I_FKP)};
Fusion_FV = concatenation (Feature Vector (PCA (I_FKP); LDA (I_FKP); ICA (I_FKP)) ;
End for
reduce FV = Feature Selection (FIS_DE (Fusion_FV));
result_KNN = result_KNN _classification (reduce FV);

2.7 Classification using KNN Classifier

The K-nearest neighbor's algorithm (KNN) is a pattern recognition method for classification and regression. The Euclidean and neural network classifier is compared to classification algorithm. KNN outperforms other classifiers in terms of classification accuracy. The KNN classifier is the best choice for classification if there is no prior information about the data distribution. In general, the KNN is a supervised classifier that is widely used to classify patterns. Furthermore, the KNN is a nonparametric classification method that completely eliminates the probability density issue. The KNN rule first classifies the optimized features x by allocating labels to the k samples, and then it makes a decision based on the label values. While the reliable parametric probability densities are unknown, the KNN classifier performs discriminant analysis. [35].

The operation of the KNN classifier is based on comparing the query image to the trained records and then determining the degree of similarity between the testing and training records. The KNN classifier uses the k training records to find new record neighbors in the search space. The closest is determined using Euclidean distance measures. Hence, the Euclidean distance is calculated using Eqs. (22) and (23).

$$X_1 = (X_{11}, X_{12}, \ldots\ldots, X_{1n}), \quad X_2 = (X_{21}, X_{22}, \ldots\ldots, X_{2n}), \tag{22}$$

$$\text{dist}(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \tag{23}$$

where x_1 and x_2 are represented as sample records with n attributes and the Euclidean distance calculates the distance between X_1 and X_2 based on the difference between the attribute values in the forms of x_1 and x_2.

3 Results and discussion

MATLAB (2015) was applied for experimental simulation with Intel®core™ i3-7100U, x-64-based processor, 64-bit operating system, and 4 GB RAM. This section presents the performance of the proposal based on some previous works such as multiple texture score level fusion [16] and hierarchical classification [36] on the PolyU finger knuckle dataset for verifying the effectiveness of the proposed work.

3.1 Performance metric

The general formula to calculate accuracy, FAR, FRR, memory, and precision is given Eqs. (24), (25), (26), (27), and (28).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100 \tag{24}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100 \tag{25}$$

$$\text{FRR} = \frac{\text{FP}}{\text{TP} + \text{FN}} \times 100 \tag{26}$$

$$\text{FAR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \times 100 \tag{27}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \tag{28}$$

where TN is specified as true negative, FN is denoted as a false negative, FP is identified as false positive, and TP is indicated as true positive.

3.2 Quantitative analysis on dataset A (12 kb)

The PolyU finger knuckle database is used to evaluate the performance of both existing and proposed work. In Table 1, with the dataset size of 12 kb, the current and proposed work performance is evaluated in terms of accuracy, FAR, recall, FRR, and precision. Here, the performance evaluation is done with 30% testing and 70% training data. Additionally, the proposed work performance is assessed with two prior classification approaches: Euclidean distance and neural network. According to the results of the experiment, the KNN classifier has a recognition accuracy of 99.67%, while the comparative classifiers (Euclidean space and neural network) have a recognition accuracy of 98.03% and 97.28%, respectively. The recall value of the KNN classifier is 98.92%, and existing classifiers deliver 97.4% and 96.90%. Successively, the precision value of KNN is 100%, and the existing classifiers achieve 94.55% and 98.76% of precision. According to the findings of the experiments, the KNN classifier outperforms other classifiers with 99.67% accuracy. The graphical depiction of the proposed work on dataset A is shown in Fig. 4.

Successively, the FRR value of KNN is 0.289%, and the existing classifiers (Euclidean distance and neural network) deliver 1.919% and 2.57% of FRR. Additionally, the FAR of the KNN classification approach is 0.361%, and the comparative classifiers achieve 2.02% and 2.87% of FAR. Table 1 shows that the DE with KNN performs significantly better than other classification approaches. The graphical depiction of the proposed work using FRR and FAR on dataset A is presented in Fig. 5.

In this section, different feature combinations are carried out with different classifiers to determine the effectiveness of the proposed work, depicted in Table 2. Here, the significance of feature extraction is analyzed with different combinations: PCA, LDA, ICA, PCA + LDA, LDA + ICA, PCA + ICA, and PCA + LDA + ICA. In Table 2, the hybrid features (PCA + LDA + ICA) with the KNN classifier attain high accuracy, recall, precision, FRR, and FAR. Existing classification methods, such as Euclidean distance and neural networks, have low accuracy, recall, precision, FRR, and FAR. It shows the KNN classification approach is superior when the training data are large and effective when dealing with noisy data.

According to the above findings, hybrid PCA_LDA feature extraction yields better results. The proposed FIS_DE selects the best subset of features and feeds it to the KNN classifier. In this research study, the optimized features determine the linear and nonlinear properties of finger knuckle regions and handle the quantitative relationships between the low-level and high-level feature vectors. The performance metrics confirm that the proposed work performs effectively well in FKP recognition. For this research, the DE is used as a mathematical model for selecting the best feature, and its parameter setting is illustrated in Table 3.

For example, for the 40 users, each user possesses 5 samples, therefore 200 images. Each test image is compared of 120 trained images with reduced feature vectors in the testing phase to evaluate whether the query image is an authorized user or not. Therefore for testing 80 images, it will take 9600 comparisons. The sample illustration of 30 runs with index is illustrated in Table 4. Table 5 shows the index value of the feature vector in Table 4 with the highest accuracy. Finally, the unique index reduced to 14 feature vectors from 73 feature vector size is shown in Table 6.

In the proposed work, the feature selection is focused on the unique feature vector. These feature vectors are selected based on the index of the feature vector with the best accuracy. The proposed work PCA LDA based on DE optimization with random feature vector selection of 10 and 5 is shown in Tables 7 and 8.

Table 1 Performance valuation of the proposed and existing works on dataset A

Classifiers	Accuracy (%)	Recall (%)	Precision (%)	FRR (%)	FAR (%)
Euclidean	98.03	97.4	94.55	1.919	2.02
Neural network	97.28	96.90	98.76	2.57	2.87
KNN	99.67	98.92	100	0.289	0.361

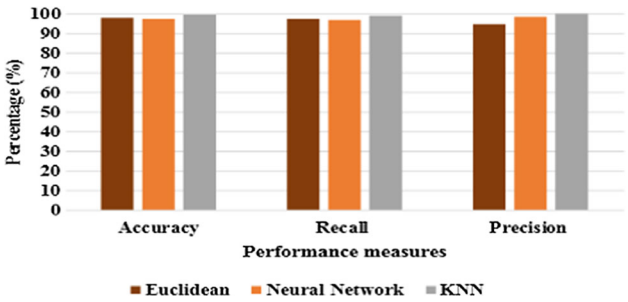

Fig. 4 Graphical depiction of the proposed work using accuracy, precision, and recall on dataset A

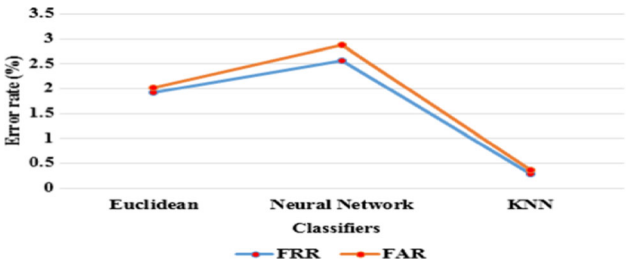

Fig. 5 Graphical depiction of the proposed work in light of FRR and FAR on dataset A

3.3 Quantitative analysis for dataset B (84 kb)

In Table 9, the proposed and existing works performances are assessed on the dataset size of 84 kb. From the experimental investigation, the recognition accuracy of KNN is 99.32%, and the current classifiers (Euclidean distance and neural network) achieve 91.83% and 96.61%. Similarly, the recall, precision, FRR, and FAR value of KNN is 97.89%, 100%, 0.628%, and 0.713%. In contrast, the comparative classification approaches (Euclidean distance and neural network) achieve minimum recall, precision, FRR, and FAR value related to the KNN classifier. The graphical depiction of the proposed work using accuracy, precision, recall, FRR, and FAR on dataset B is denoted in Figs. 6 and 7.

As discussed in the proposed work, feature extraction and optimization are an integral part of finger knuckle-based recognition. In this research, several feature values are extracted under feature extraction using PCA, ICA, and LDA. After extracting the features, optimization is crucial to reduce the curse of dimensionality issue, and also, the optimized features are fit for the better classification. The dataset B with image size 84 kb acquired more time in the training phase, and the number of features is also increased compared with dataset A. The proposed work achieves 99.67% and 99.32% accuracies for both datasets A and B,

Table 2 Evaluation of the proposed work with different feature combinations on dataset A

Classifiers	PCA	ICA	LDA	PCA + LDA	LDA + ICA	PCA + ICA	PCA + LDA + ICA
Accuracy (%)							
Euclidean	72.55	60.54	75.30	98.03	92	96.69	83.59
Neural network	74.23	65.35	76	97.28	93.43	95.15	84.23
KNN	81.65	80.78	88.13	99.67	95.68	98.05	95.48
Recall (%)							
Euclidean	65.91	54.91	76.20	97.4	87.87	89.24	76.92
Neural network	49	55.25	33.34	96.9	69	72.04	44.90
KNN	73.78	66.93	63.09	98.92	86.05	92.16	70.20
Precision (%)							
Euclidean	41.01	67.93	55.11	94.55	78.82	87.74	48.95
Neural Network	52.09	55.87	66.33	98.76	74.03	81.48	64.01
KNN	77.03	79	73.02	100	82.52	89.07	61.63
FRR (%)							
Euclidean	33.98	40.98	20.87	1.919	8.07	3.34	18.90
Neural network	33.71	39.32	19.03	2.57	6.06	4.81	13.09
KNN	28.67	21.09	12.67	0.289	2.96	1.97	5.05
FAR (%)							
Euclidean	20.91	37.94	28.52	2.02	7.93	3.28	13.92
Neural network	17.84	29.99	29.97	2.87	7.08	4.88	18.44
KNN	8.02	17.35	11.07	0.361	5.67	2.02	4

Table 3 DE parameter setting

Parameters	Size
Population size (NP)	50
Scaling factor (F)	0.85
Cross over (CR)	0.7
Mutation (Pmu)	0.5
Max_iteration	100
Independent number of runs	30

Table 4 Sample of 30 runs with best index of feature vector based on accuracy

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	Accuracy (%)
1	5	6	7	9	10	18	37	63	65	0.98
6	9	15	17	18	20	28	35	38	40	0.98
5	17	18	22	32	36	39	40	59	72	0.98
2	6	23	25	33	39	44	45	46	64	0.97
1	5	7	8	13	16	32	38	43	72	1
1	5	13	18	21	35	37	46	55	70	0.98
1	3	5	12	13	16	18	21	56	60	0.98
1	3	5	7	0	20	21	33	39	70	1
12	21	26	29	38	51	54	57	59	65	0.98
1	7	9	13	24	25	37	40	55	57	1
1	6	7	24	33	38	41	43	48	58	0.98
5	17	20	23	27	29	45	50	63	71	0.98
1	9	19	24	26	30	35	39	64	73	0.97
4	9	14	25	34	38	52	56	57	64	0.98
2	5	6	12	17	22	29	41	55	62	0.98
5	11	13	16	17	20	23	38	46	73	1
1	5	9	21	28	30	34	39	65	70	1
5	6	13	15	17	18	21	42	43	53	0.98
8	15	18	20	21	24	31	32	54	63	1
10	11	18	21	28	33	46	51	59	65	0.98
1	6	13	14	16	35	44	45	52	69	1
1	4	9	14	20	21	24	32	33	44	1
2	10	12	14	18	20	24	28	31	33	0.98
1	5	8	18	26	30	31	32	38	57	0.98
7	12	13	17	22	39	40	44	51	55	0.98
1	4	5	17	21	30	42	47	55	58	0.98
1	17	20	38	39	42	47	50	54	59	0.98
5	7	12	17	20	23	26	49	52	57	1
4	6	17	21	24	33	35	36	50	59	1
4	9	13	17	21	24	26	28	44	72	0.97

respectively. Table 10 and 11 indicate the proposed work PCA_LDA based on DE optimization random feature vector selection of 10 and 5, respectively.

Table 5 Selected index subset feature vectors with the highest accuracy

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
1	5	7	8	13	16	32	38	43	72
1	3	5	7	17	20	21	33	39	70
1	7	9	13	24	25	37	40	55	57
5	11	13	16	17	20	23	38	46	73
1	5	9	21	28	30	34	39	65	70
8	15	18	20	21	24	31	32	54	63
1	6	13	14	16	35	44	45	52	69
1	4	9	14	20	21	24	32	33	44
5	7	12	17	20	23	26	49	52	57
4	6	17	21	24	33	35	36	50	59
4	9	13	17	21	24	26	28	44	72

Table 6 Selected subset feature vectors with unique index position

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14
1	4	5	7	9	13	16	17	20	21	24	32	33	44

The effectiveness of feature optimization is shown in Table 12. In this scenario, the performance of the proposed work is verified by utilizing the performance metric (accuracy). Under this circumstance, the accuracy of DE-based feature optimization is 22% better than other techniques on dataset B. Additionally, the individual features show low accuracy in finger knuckle-based recognition related to hybrid features. Hence, the hybrid feature extraction improved accuracy in finger knuckle-based recognition up to 2–7% about individual characteristics.

3.4 Comparative study

The performance of the proposed and existing works is compared in Table 13. By combining multiple texture attribute descriptors; Nigam et al. [16] created a new FKP-based recognition method. For estimating the image quality of FKP images, authors used both general and trait-specific quality parameters. On the PolyU finger knuckle dataset, this paper validated the efficacy of the developed work. The practical result shows that the developed work obtained a recognition accuracy of 99.28%. Kong et al. [36] developed a new hierarchical classification approach for FKP-based recognition. The feature vectors were extracted from the FKP images using Gabor features at first. Then, using a pre-defined threshold, create a new decision rule. Finally, the speeded-up robust feature was

Table 7 Evaluation of the proposed work PCA_LDA with DE optimization based on features on dataset A

No. of runs	No. of iterations	Reduced number of features	Recognition accuracy (%)
30	3000	14	99.67
25	2500	32	89.13
20	2000	42	98.25
15	1500	32	100
10	1000	39	99.89
5	500	35	100

Table 8 Evaluation of the proposed work based on random selection of feature vector: 5

No. of runs	No. of iterations	Reduced number of features	Recognition accuracy (%)
30	**3000**	13	98.65
25	2500	10	98.65
20	2000	5	97.32
15	1500	5	95.94
10	1000	5	94.59
5	500	5	97.3

Table 9 Performance investigation of the existing and proposed work on dataset B

Classifiers	Accuracy (%)	Recall (%)	Precision (%)	FRR (%)	FAR (%)
Euclidean	91.83	86.53	90.60	7.78	8.56
Neural Network	96.61	91.11	100	2.88	3.90
KNN	99.32	97.89	100	0.628	0.713

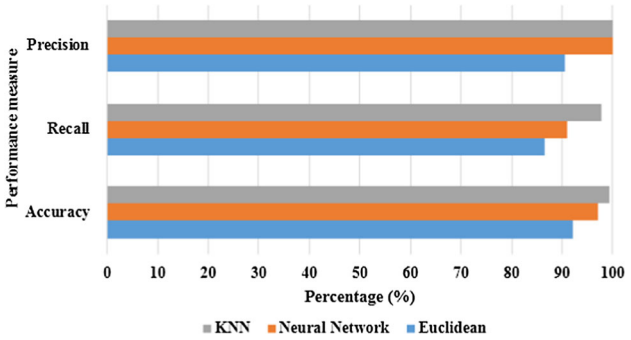

Fig. 6 Graphical depiction of the proposed work using accuracy, precision, and recall on dataset B

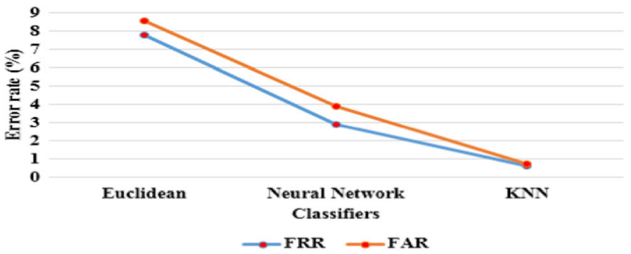

Fig. 7 Graphical depiction of the proposed work through FRR and FAR on dataset B

used for authentication. The experimental investigation was performed on the PolyU finger knuckle dataset, and the developed system attained 98.5% of recognition accuracy. The Curvelet transform is used in [37] to decompose the knuckle images into sub bands, and PCA is used for feature extraction. The accuracy of the experimental results is 98.72%. The proposed work achieved 99.67% of recognition accuracy on datasets A and achieved 99.32% for dataset B, which were higher related to the existing works. In the proposed work, feature optimization is a fundamental part of FKP-based authentication. Every FKP image comprises numerous features that lead to the "curse of dimensionality" issue. Thus, feature optimization is crucial for optimizing the feature vectors that are suitable for better classification.

4 Conclusion

The most practical and cost-effective biometric authentication method is the finger knuckle print. PCA LDA is used to extract hybrid features in this analysis. The image's irrelevant and redundant information reduces accuracy and requires more processing time. Hence, the optimal subset of features is selected from the FKP images. The proposed fitness index selection with differential evolution (FIS_DE) algorithm is used to select the relevant features. From the simulation, the proposed FIS_DE with KNN achieved 99.67% of accuracy for dataset A and 99.32% for dataset B

Table 10 Evaluation of the proposed work PCA_LDA with DE optimization based on features on dataset B

No. of runs	No. of iterations	Reduced number of features (10)	Recognition accuracy (%)
35	3500	43	98.65
30	3000	16	99.32
25	2500	34	94
20	2000	45	100
15	1500	36	100
10	1000	41	100
5	500	31	100

Table 11 Evaluation of the proposed work PCA_LDA with DE optimization based on features on dataset B

No. of runs	No. of iterations	Reduced number of features (5)	Recognition accuracy (%)
35	3500	10	97.5
30	3000	15	98.6
25	2500	14	96.65
20	2000	7	97.32
15	1500	8	95.94
10	1000	8	94.59
5	500	8	95.3

Table 12 Evaluation of the proposed work with dissimilar feature combinations and optimizations on dataset B

Recognition accuracy (%)								
Optimizer	Classifiers	PCA	ICA	LDA	PCA + LDA	LDA + ICA	PCA + ICA	PCA + LDA + ICA
Without optimizer	Euclidean	50.98	42.09	58.03	67.42	70.98	52.22	68
	Neural network	60.87	38.88	60.06	68.68	55.54	64.28	68.03
	KNN	60.16	41.01	62.33	68.90	66.67	50	70.08
DE	Euclidean	87.18	78.9	81.2	92.2	60	60	90.2
	Neural network	38.46	81.8	80.34	97.12	80.53	82.46	92.12
	KNN	69.23	78.9	82.91	99.32	66.04	40.35	94.07

Table 13 Comparative study of the proposed and existing systems

Methodology	Dataset	Recognition accuracy (%)	
Geometric and Texture analysis [37]	PolyU finger knuckle dataset	98.72	
Multiple texture score level fusion [16]		99.28	
Hierarchical classification [36]		98.5	
Proposed system (DE-KNN)		Dataset A	99.67
		Dataset B	99.32

that are higher related to the existing works. The number of features in the proposed work is reduced to 14 for dataset A and 16 for dataset B. Each FKP image takes 5 s to train with the size of 84 kb, while dataset A takes 2 s. Therefore the size of the image plays a vital role in consuming the memory and time. To improve the FKP-based recognition performance even more, an effective deep learning-based supervised system will be developed in the future.

Declarations

Conflict of interest We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, and there is no professional or other personal interest of any nature or kind in any product, service and company.

Human and animal rights statement We are not used any animals and human for our research. The manuscript not submitted anywhere for publications and all work mentioned in this paper is authors own

work. We did not use any third-party software for surveys, scales, and studies.

References

1. Nigam A, Gupta P (2015) Designing an accurate hand biometric based authentication system fusing finger knuckleprint and palmprint. Neurocomputing 151:1120–1132
2. Muthukumar A, Kavipriya A (2019) A biometric system based on Gabor feature extraction with SVM classifier for Finger-Knuckle-Print. Pattern Recogn Lett 125:150–156
3. Hanmandlu M, Grover J (2012) Feature selection for finger knuckle print-based multimodal biometric system. Int J Comput Appl 38(975–8887):2
4. Kong T, Yang G, Yang L (2014) A new finger-knuckle-print ROI extraction method based on probabilistic region growing algorithm. Int J Mach Learn Cybern 5:569–578
5. Zhang L, Zhang L, Zhang D, Guo Z (2012) Phase congruency induced local features for finger-knuckle-print recognition. Pattern Recogn 45:2522–2531
6. Zhang L, Zhang L, Zhang D, Zhu H (2011) Ensemble of local and global information for finger–knuckle-print recognition. Pattern Recogn 44:1990–1998
7. Shariatmadar ZS, Faez K (2014) Finger-Knuckle-Print recognition performance improvement via multi-instance fusion at the score level. Optik Int J Light Electron Optics 125:908–910
8. Ozkaya N, Kurat N (2014) Discriminative common vector based finger knuckle recognition. J Vis Commun Image Represent 25:1647–1675
9. Morales A, Travieso CM, Ferrer MA, Alonso JB (2011) Improved finger-knuckle-print authentication based on orientation enhancement. Electron Lett 47:380–381
10. Gao G, Zhang L, Yang J, Zhang L, Zhang D (2013) Reconstruction based finger-knuckle-print verification with score level adaptive binary fusion. IEEE Trans Image Process 22:5050–5062
11. Hegde C, Shenoy PD, Venugopal KR, Patnaik LM (2013) Authentication using finger knuckle prints. Signal Image Video Process 7:633–645
12. Chlaoua R, Meraoumia A, Aiadi KE, Korichi M (2019) Deep learning for finger-knuckle-print identification system based on PCANet and SVM classifier. Evol Syst 10:261–272
13. Attia A, Chaa M, Akhtar Z, Chahir Y (2018) Finger knuckle patterns based person recognition via bank of multi-scale binarized statistical texture features. Evol Syst 11:1–11
14. Kumar A, Wang B (2015) Recovering and matching minutiae patterns from finger knuckle images. Pattern Recogn Lett 68:361–367
15. Usha K, Ezhilarasan M (2015) Contourlet transform based feature extraction method for finger knuckle recognition system. In Comput Intell Data Min 3:407–416
16. Nigam A, Tiwari K, Gupta P (2016) Multiple texture information fusion for finger-knuckle-print authentication system. Neurocomputing 188:190–205
17. Gao G, Yang J, Qian J, Zhang L (2014) Integration of multiple orientation and texture information for finger-knuckle-print verification. Neurocomputing 135:180–191
18. Grover J, Hanmandlu M (2015) Hybrid fusion of score level and adaptive fuzzy decision level fusions for the finger-knuckle-print based authentication. Appl Soft Comput 31:1–13
19. Aoyama S, Ito K, Aoki T (2014) A finger-knuckle-print recognition algorithm using phase-based local block matching. Inf Sci 268:53–64
20. Agrawal P, Ganesh T, Mohamed AW (2020) A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection. Neural Computing and Applications 33(11):5989–6008. https://doi.org/10.1007/s00521-020-05375-8
21. Ghoualmi L, Draa A, Chikhi S (2015) An efficient feature selection scheme based on genetic algorithm for ear biometrics authentication. In: 12th International symposium on programming and systems (ISPS)m, pp 1–5. https://doi.org/10.1109/ISPS.2015.7244991
22. Ghoualmi L, Draa A, Chikhi S (2016) An ear biometric system based on artificial bees and the scale invariant feature transform. Expert Syst Appl 57:49–61
23. Alpaslan Altun A, Erdinc Kocer H, Allahverdi N (2008) Genetic algorithm based feature selection level fusion using fingerprint and iris biometrics. Int J Pattern Recognit Artif Intell 22:1–16
24. Sasirekha K, Thangavel K (2019) Optimization of K-nearest neighbour using particle swarm optimization for face recognition. Neural Comput Appl 31:7935–7944
25. Agrawal P, Abutarboush HF, Ganesh T, Mohamed AW (2021) Metaheuristic algorithms on feature selection: a survey of one decade of research (2009–2019). IEEE Access 9:26766–26791
26. Moradi P, Gholampour M (2016) A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. Appl Soft Comput 43:117–130
27. Reyhaneh Yaghobzadeh, Seyyed Reza Kamel, Mojtaba Asgari, & Hassan Saadatmand (2020) A Binary Grasshopper Optimization Algorithm for Feature Selection. Int J Eng Res Technol 09(03). https://doi.org/10.17577/ijertv9is030420
28. Palisetty AK, Gogulamanda JS (2021) A robust method for multi-algorithmic palmprint recognition using exponential genetic algorithm-based feature selection. communication software and networks. Lecture Notes in Networks and Systems, 134 Springer, Singapore
29. Larabi-Marie-Sainte S, Ghouzali S (2020) Multi-objective particle swarm optimization-based feature selection for face recognition. Stud Inf Control 29:99–109
30. Khushaba R, Al-Ani A & Al-Jumaily A (2009) Differential evolution based feature subset selection, pp 1–4. https://doi.org/10.1109/ICPR.2008.4761255
31. Maheshwari, Radhika & Kumar, Manoj & Kumar, Sushil. (2016). Optimization of Feature Selection in Face Recognition System Using Differential Evolution and Genetic Algorithm. https://doi.org/10.1007/978-981-10-0451-3
32. Jaswal G, Poonia RC (2020) Selection of optimized features for fusion of palm print and finger knuckle-based person authentication. Expert Sys 38(1). https://doi.org/10.1111/exsy.12523
33. Kumar A (2014) Importance of being unique from finger dorsal patterns: Exploring minor finger knuckle patterns in verifying human identities. IEEE Trans Inf Forensics Secur 9(8):1288–1298
34. Dataset link: https://www4.comp.polyu.edu.hk/~csajaykr/fn1.h
35. Cover T, Hart P (1967) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13:21–27. https://doi.org/10.1109/TIT.1967.1053964
36. Kong T, Yang G, Yang L (2014) A hierarchical classification method for finger knuckle print recognition. EURASIP J Adv Signal Process 2014:44
37. Usha K, Ezhilarasan M (2016) Personal recognition using finger knuckle shape oriented features and texture analysis. J King Saud Univ Comput Inf Sci 28:416–431