Journal of Engineering Science and Technology Review 3 (1) (2010) 70-79

JOURNAL OF
Engineering Science and
Technology Review

www.jestr.org

Review Article

Evolutionary Algorithms Application Analysis in Biometric Systems

N. Goranin[*] and A. Cenys

Information Security Laboratory, Department of Information Systems, Faculty of Fundamental Sciences, Vilnius Gediminas Technical University Sauletekio al. 11, SRL-I-415, LT-10223, Vilnius, Lithuania.

Received 19 February 2010; Accepted 13 March 2010

Abstract

Wide usage of biometric information for person identity verification purposes, terrorist acts prevention measures and authentication process simplification in computer systems has raised significant attention to reliability and efficiency of biometric systems. Modern biometric systems still face many reliability and efficiency related issues such as reference database search speed, errors while recognizing of biometric information or automating biometric feature extraction. Current scientific investigations show that application of evolutionary algorithms may significantly improve biometric systems. In this article we provide a comprehensive review of main scientific research done in sphere of evolutionary algorithm application for biometric system parameter improvement.

Keywords: genetic, algorithm, evolutionary, biometry.

1. Introduction

In today's world there is a growing concern regarding identity theft, national security, and on-line terrorism [1]. Biometrics are seen by many researchers as a solution to a lot of user identification and security problems now a days [2]. Biometric identification is any automatically measurable, robust and distinctive physical characteristic or personal trait that can be used to identify an individual or verify the claimed identify of an individual [3]. Biometric science utilizes the measurements of a person's behavioral characteristics (keyboard strokes, mouse movement) or biological characteristics (fingerprint, iris, nose, eyes, jaw, voice pattern, etc). It is these measurements that are then used to create a reference template, which the recognition software uses to identify or authorize an individual as the person they claim to be [1]. The most common biometric method to identify the individuals is through fingerprint [4] recognition. In recent years, there has been significant interest in using other biometrics for identifying individuals. These include techniques that rely on: DNA, hand geometry, palm print, face (both optical and infrared), iris, retina, signature, ear shape, odor, keystroke entry pattern, gait, and voice [5]. Other emerging biometrics such as ear force fields [6], heart signals [7], and electroencephalogram (EEG) or brain signals [8] have also been proposed in recent years. Smart identity cards store unique biometric informatics such as the fingerprint, signature, retinal pattern, voice recognition and facial features [9].

Biometric system implementation for person identity verification purposes, terrorist acts prevention measures, authentication process simplification in computer systems and many other tasks has raised significant attention to reliability and efficiency of biometric systems. Modern biometric systems still face many reliability and efficiency related issues such as reference database search speed, errors while recognizing of biometric information or automating biometric feature extraction. These problems will be discussed later in more detail when discussing specific biometric techniques. Scientific investigations show that application of evolutionary algorithms may significantly improve biometric systems. That is why it is necessary to get an understanding of evolutionary algorithms and current research done in this area for practical improvement of biometric system quality.

Evolutionary algorithms are inspired by Darwinian evolution mechanisms which include reproduction, mutation, recombination, and selection. Genetic Algorithms (GA) which are considered to be a part of Evolutionary algorithms, could mimic nature to computationally emulate the same 'survival of the fittest' paradigm for difficult problems. GAs [10] are nondeterministic methods that employ crossover and mutation operators for deriving offsprings. GAs are fully defined when one provides the strategy for deriving the offsprings and the composition of the next generation. Simulated breeding is the usual strategy used when offsprings are selected according to their fitness [11]. GAs work by maintaining a constant-sized population of candidate solutions known as individuals ('chromosomes'). The power of a GA lies in its ability to exploit, in a highly efficient manner, information

* E-mail address: ngrnn@fmf.vgtu.lt
ISSN: 1791-2377

about a large number of individuals [11]. Other advantage GA gives are [12], [13]:

- no prior assumptions about the solution are needed;
- it works with a coding of parameter sets and not with the parameters themselves;
- algorithm searches from a population of candidate solutions, not a single point;
- GA uses objective function information, not derivative or other auxiliary information;
- it uses probabilistic transition rules, not deterministic rules;
- the result is a population of solutions instead of an individual solution.

The search underlying GAs is such that breadth and depth – exploration and exploitation - are balanced according to the observed performance ('fitness') of the individuals evolved so far. By allocating more reproductive occurrences to above average individual solutions, the overall effect is to increase the population's average fitness [11]. The general algorithm flow is presented in Figure 1.

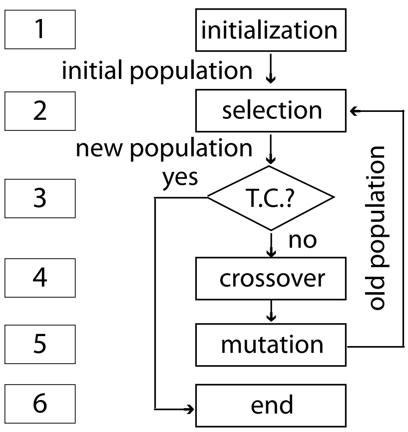

Figure 1. GA flow chart [14]

During the initialization stage (Figure 1-1) initial population of solutions is generated. Each solution is represented as a chromosome. At selection stage (Figure 1-2) solutions are selected (individual genomes are chosen from a population for later breeding (recombination or crossover)) through a fitness-based process and in case termination condition (T.C., Figure 1-3) is not met (e.g. number of generation or sufficient fitness achieved) evolutionary mechanisms (crossover - genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next; mutation - genetic operator used to maintain genetic diversity from one generation of a population of chromosomes to the next) are started (Figure 1-4/5). In case termination condition is reached, algorithm execution is ended (Figure 1-6).

Many crossover techniques exist for organisms which use different data structures to store themselves (Figure 2).

The mutation operator involves a probability that an arbitrary bit in a genetic sequence will be changed from its original state. A common method of implementing the mutation operator involves generating a random variable for each bit in a sequence (Figure 3).

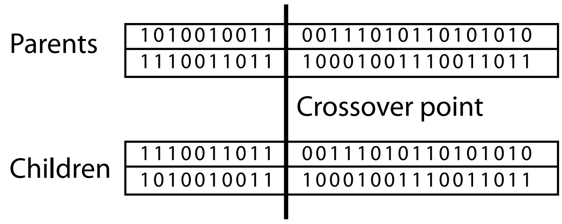

Figure 2. Single point crossover

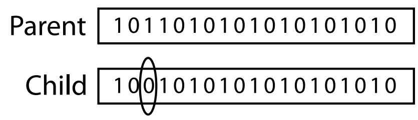

Figure 3. Mutation in GA

Genetic programming (GP) – one more representative of Evolutionary algorithms, is used for automated learning of computer programs. The most commonly used representation in genetic programming is the program tree (Figure 4) [15].

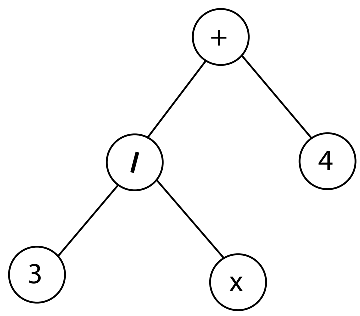

Figure 4. Basic tree-like program representation used in GP [15]

When applying GP to a problem, there are five major preparatory steps to be determined [16], [17]:

1. The set of terminals (e.g., the independent variables of the problem, zero-argument functions, and random constant) for each branch of the to-be-evolved program.
2. The set of primitive functions (e.g., Boolean functions, arithmetic functions, conditional functions) for each branch of the to-be-evolved program.
3. The fitness measure (for explicitly or implicitly measuring the fitness of individuals in the population).
4. Certain parameters for controlling the run. And
5. The termination criterion and method for designating the result of the run. Genetic operators.

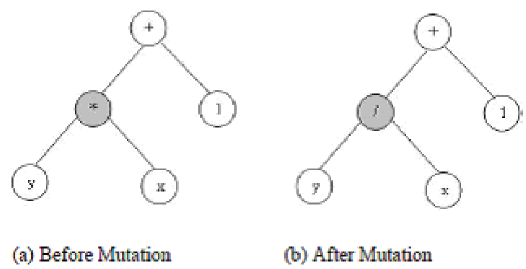

Figure 5. Mutation in genetic programming [15]

GP also uses mutation (randomly selects a node within the parent tree as the mutation point, generates a new tree of maximum depth and replaces the subtree rooted at the selected node with the generated tree, Figure 5) and crossover (randomly selects a node within each tree as crossover points, takes the sub tree rooted at the selected node in the second parent and uses it to replace the sub tree rooted at the selected node in the first parent to generate a child (and optionally does the reverse to obtain a second child), Figure 6) operators.

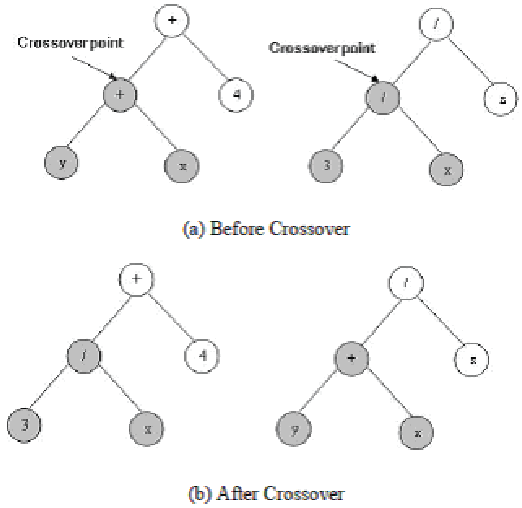

Figure 6. Crossover in genetic programming [15]

In summary, genetic programming creates computer programs by executing several steps [15] dedicated for specific problem solution.

2. Fingerprints

Fingerprint has been researched the longest period of time and shows the most promising future in real-world applications [18]. A fingerprint is formed of a group of curves. The most common characteristics include endpoints and bifurcations called as minutiae [19]. Due to the persistence and individuality of fingerprints, fingerprint recognition has become a popular personal identification technique [20]. However, because of the complex distortions among the different impressions of the same finger, fingerprint recognition is still a challenging problem [18]. Other real world problems related with fingerprint usage in biometric systems are fingerprint registration [18], creation and usage of fingerprint test databases [21] and many others.

2.1 GA application for fingerprint registration

According to [18] image registration algorithms fall within two categories: area based methods and feature based methods. The original image is often referred to as the reference image and the image to be mapped onto the reference image is referred to as the target image. For area based image registration methods, the algorithm looks at the structure of the image via correlation metrics, Fourier properties and other means of structural analysis. Feature based methods, tune their mappings to the correlation of image features: lines, curves, points, line intersections, boundaries, etc. Image registration algorithms can also be classified according to the transformation model used to relate the reference image space with the target image space. The first category of transformation models includes linear transformations (Figure 7), which are a combination of translation, rotation, global scaling, shear and perspective components. Linear transformations are global in nature, thus not being able to model local deformations. The second category includes 'elastic' or 'nonrigid' transformations. These transformations allow local warping of image features, thus providing support for local deformations. Nonrigid transformation approaches include polynomial wrapping, interpolation of smooth basis functions (thin-plate splines and wavelets) and physical continuum models.

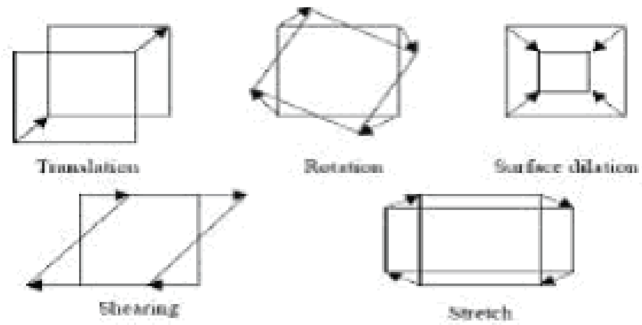

Figure 7. Elementary Geometric Transforms for Planar Surface Elements [18]

In paper [18] GA is used for registration of two images. Two images can not be compared if one of these images is translated or rotated by some unknown angle. To obtain the registration, the first image must be transformed until gets best matching with the second image. However, the conventional registration methods suffer from mis-registration because of the difference in rotation angles. Authors investigate the use of GA in image registration, since the GA is accurate, very fast and now is used very frequently. Image registration is described as the process of overlaying two or more images of the same scene taken at different times, from different viewpoints and/or by different sensors. Search-based image registration methods utilize an iterative procedure to improve the initial guess of the unknown transform parameters. Registration process mainly consist of determining the unknown transformation parameters required to map the input image to the reference image in order to compare and analyze both in a common reference frame.

Automatic registration of the two images requires determining transformation applied and a measure of similarity. The applied transformation is given by affine transformation where an affine transform is a linear coordinate transformation that includes the elementary transformations. The affine transform can be expressed by vector addition and matrix multiplication as:

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (1)$$

The affine transform can be rewritten by the following:

$$X = t_x + a_{11}x + a_{12}y \quad (2)$$

$$Y = t_y + a_{21}x + a_{22}y \tag{3}$$

where x, y are the two Cartesian spatial dimensions and a_{ij} and t_x, t_y are the adjustable parameters whose values are to be estimated under the following constraints: -size of image = t_x*t_y= size of image and -1=a_{ij}= 1.

The measure of fitness, or success of the transformation is based simply on the point by point absolute difference between the two images

$$f = 1/m \bullet_{j=1}^{m} \left|\bullet c(j)\right| \tag{4}$$

where m-number of points considered, *c -grayscale intensity difference between the same point in the first and the transformed second image. Since the two images may well have different overall intensity distributions an additional unknown parameter a_{31} is required to equalized the distributions

$$\bullet c(j) = c_1(j) - a_{31}c_2(j) \tag{5}$$

where $c_1(j)$ and $c_2(j)$ are the individual grayscale intensities of pixel j for the two images (a_{31} is the same for all j).

The similarity measure provides a quality index of each solution. The choice of the similarity measure is closely related to the selected feature since it measures the similarity between same features in the reference and the transformed input image typically, similarity measures are the correlation. The sum of absolute differences, the root mean square. The normalized cross-correlation function (CC) can be written as:

$$CC = \frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left(\left(R_{ij} - R\right)\times\left(I_{ij} - I\right)\right)}{\sqrt{\sum\sum\left(R_{ij} - R\right)^2 \times \sum\sum\left(I_{ij} - I\right)^2}} \tag{6}$$

The technique was applied to get some of rotated figures with some known rotation angles. Authors state that they could retrieve the rotation angles successfully up to an accuracy approaching 100% (Percentage relative error: 0-10%). The time of execution was reasonable. It was noticed that the wider the rotation angle is, the less the relative error will be i.e. the rotation angle is inversely proportional to the relative error. The method can be suitable for fingerprints registration in large databases. This recommendation comes from the fact that the time consumed in finding the rotation angle is so minute.

2.2 GA and GP application for fingerprint matching

2.2.1 General overview

Fingerprint matching depends on the comparison of the characteristic of local ridges and their relationships. Most of the existing automatic fingerprint verification systems are based on minutiae features (ridge bifurcation and ending). Such systems first detect the minutiae (Figure 8) in a fingerprint image and then match the input minutiae set with the stored template [21], [22], [23]. Extracting minutiae from fingerprint images is one of the most important steps in automatic fingerprint identification system. A number of publications is available on the topic GA and GP application for fingerprint matching.

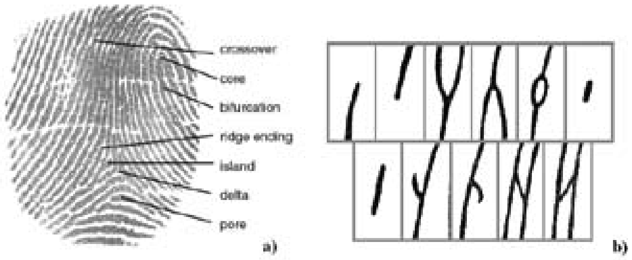

Figure 8. a) Fingerprint sample with minutiae marked [24], b) minutiae types [25]

Authors of [26] propose a fingerprint matching approach based on GA, which finds the optimal global transformation between two different fingerprints. In order to deal with low quality fingerprint images, which introduce significant occlusion and clutter of minutiae features, they design the fitness function based on the local properties of each triplet of minutiae. The experimental results on National Institute of Standards and Technology fingerprint database, NIST-4, not only show that the proposed approach can achieve good performance even when a large portion of fingerprints in the database are of poor quality, but also show that the proposed approach is better than another approach, which is based on mean-squared error estimation.

In [27] parameter optimization for biometric fingerprint recognition using GA is discussed. The created application is planned so that it can be used without great effort for different biometric systems. Instead of estimating the required parameters as in the case of some methods, here they are determined with the help of GA. The test database used consisted of 1200 fingerprints of 12 persons. For the confirmation of the results, which were found out with this test set, the databases of the Fingerprint Verification Contests of the years 2000, 2002 and 2004 were examined in addition. In the best case an improvement in the recognition performance of 38% was observed.

A Kohonen self-organizing neural network embedded with GA for fingerprint recognition was proposed in [28]. The GA was embedded to initiate the Kohonen classifers. By the proposed approach, the neural network learning performance and accuracy were greatly enhanced. In addition, the GA could successfully avoid the neural network from being trapped in a local minimum. The proposed method was tested for the recognition of fingerprints. The results were promising to applications.

[19] presents a genetic algorithm aimed at optimizing the transformation between two sets of minutiae extracted from two different fingerprints belonging to the same finger. Experiments with FVC2004 image database were performed. Based on preliminary results, one may conclude that the system developed in this work obtained good accuracy rate in the verification process, considering the restriction variation imposed by the threshold.

While most fingerprint matching systems rely on the distribution of minutiae on the fingertip to represent and match fingerprints and the ridge flow pattern is generally used for classifying fingerprints, it is seldom used for matching. The [29] article describes a new method for fingerprint matching based on lines extraction and graph matching principles.

2.2.2 GP application for minutiae points extraction

Since it is not possible to analyze all GA and GP application for fingerprint matching cases in one article and taking into consideration that GA mechanisms were described in detail in section 2.1 here as a an example we provide technical analysis of [15] were GP techniques are used to extract mathematical formulas for minutiae points (end points and bifurcation points).

At first the fingerprint images are enhanced and filtered using enhancing and filtering technique. After that the minutiae points are extracted from the enhanced image. Figure 9(a) shows the query fingerprint, figure 9(b) is the fingerprint after enhancement and filtering, figure 9(c) is the fingerprint image with the determined minutiae points shown in the figure.

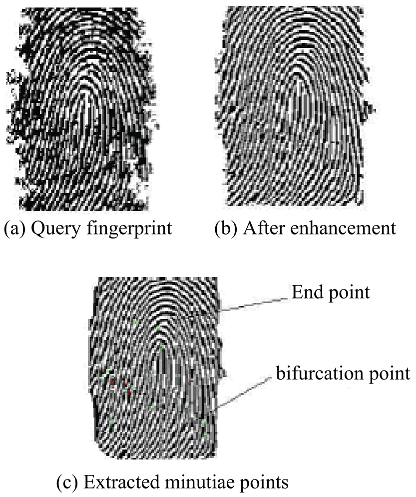

Figure 9. Query fingerprint image, after enhancement and extracted minutiae points [15]

After that two graphs can be drawn describing these points (Figure 10).

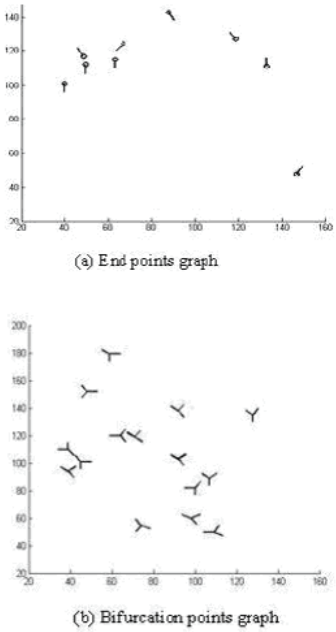

Figure 10. Two graphs explain minutiae points [15]

This information is used to develop two mathematical formulas describing the relationship between the minutiae points using genetic programming. Figure 11a shows part of the mathematical formula for end points in S-expression, and figure 11b shows part of the mathematical formula for bifurcation points after running the GP algorithm.

```
(+ (- (% (+ (% (% -8 X) (*(% (+ (% ANGLE X) -8) -2) ANGLE))
      (% (% ANGLE -1) (% (+ -9 ANGLE) (- 3 -2))))
    (- (+ (% (+ (+ 7 ANGLE) -2) X) (- -5 2))
      (- (% -5 (* X -4)) (+ ANGLE 9))))
  (- (*(% (- X ANGLE) (+ X - 1))
      (- (% (+ (+ X (+ -2 (- (% X ANGLE) ANGLE))) X) -6)
        (*(% (- X ANGLE) (+ (* ANGLE X) -1))
          (- (% (% (+ ANGLE X)
                (*(% (*(*10 (+ ANGLE (+ 1 -3)))
                      (% (*ANGLE ANGLE) 2))
                    (+ (+ 9 -3)
                      (% (- (+ (*(+ X X) X))))))
.
.
.
```

(a) End Points formula

```
(+ (% (- (+ (% (*10 (% (% (* -9 ANGLE2) ANGLE3) -2)) (% ANGLE3 -
        (*(*(- ANGLE3 3) (+ ANGLE2 ANGLE1))
          (+ (- -7 ANGLE3) (- ANGLE2 3))))
      (+ (- (- 10 -4) (*-7 ANGLE3))
        (-9
          (% (*(*6 -8) ANGLE3)
            (% (- (+ (% (*10 (% (% (*-9 ANGLE2) ANGLE3) -2))
                    (% ANGLE3 -1))
                  (*(*(- ANGLE3 3) (+ ANGLE2 ANGLE1))
                    (+ (- ANGLE2 3) (- ANGLE2 3))))
                (+(- (- 10 -4) (*-7 ANGLE3))
                  (% (- (% (* ANGLE3 -3)
                        (- (+ (% ANGLE1
                              (+ (- (- (+ ANGLE3
                                    (% (- (% (- (+ (*(+ (% (+ (% X
                                                          3)
                                                        (% (% (% (- -5
                                                                X)
                                                              (% ANGLE2
                                                                6))
                                                            (- 2
                                                              (- ANGLE1
                                                                -5)))
                                                          3))
                                                      -3)
                                                    2)))))))
.
.
.
```

(b) Bifurcation points formula

Figure 11. Part of mathematical formula for minutiae points (S Expression) [15]

After these formulas are received, the query fingerprint is compared with all stored fingerprints to find a match to the query fingerprint. In case three fingerprints are stored with the minutiae points, then each of these points is applied in the mathematical formulas of the query fingerprint. The points that solve for y and is equal to the result, y, of the query fingerprint is a matching image. The results must be equal in end points formula and bifurcation points formula, otherwise there is no match.

2.3 GA application for fingerprints image generation

Constructing a fingerprint database is important to evaluate the performance of automatic fingerprint recognition systems. The construction of fingerprint databases requires an enormous effort so that, in practice, it is often too costly or the resulting database is incomplete or unrealistic [30]. Because of the difficulty in collecting fingerprint samples, there are only few benchmark databases available. Moreover, various types of fingerprints are required to measure how robust the system is in various environments. [31] presents a method that generates various fingerprint images automatically from only a few training samples by using the GA. Fingerprint images generated by the proposed method include similar characteristics of those collected from a corresponding real environment.

When a target environment is given, the proposed method constructs a set of filters that modifies an original image so as to become similar to that collected in the environment. A proper set of filters is found by the GA, where fitness evaluation is conducted using various statistics of fingerprints to measure the similarity. In the initialization step, it sets basic parameters including the population size, the maximum number of generations, the length of chromosomes, selection strategy, selection rate, crossover rate and mutation rate. The length of chromosomes means the size of a filter composed, where each gene in the chromosome represents the corresponding filter in the pool of filters. Only a few samples are required to calculate several statistics for the target environment to evaluate a chromosome. The fitness of a chromosome is obtained from the similarity between a few real images from the target environment and images generated after filtering. The value of each gene means a filter to apply for images of the training database.

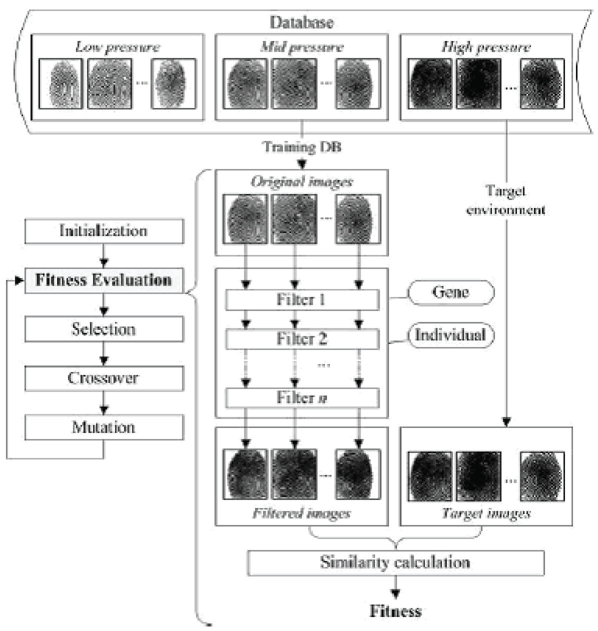

Figure 12. Fingerprint image generation method [31]

Popular image filters (Brightness (3 values), Contrast (3 values), Stretch, Equalize, Logarithm, Blur (6 masks), Sharper (4 masks), Median (10 masks), Morphology(10 masks) Erosion, Dilation, Opening, Closing) are used to produce similar effects in real environments. The order and type of filters used in the filter set are determined by the GA, because it is practically impossible to test all the cases of composition. The fitness of a filter set is estimated by measuring the similarity between fingerprints collected from the target environment and images generated by the composite filter. Several representative features of fingerprints, such as the mean and variance of images, directional contrasts, average ridge thickness and interval, singularities and minutiae, are used to design the fitness evaluation function, in which weights are heuristically determined:

$$\begin{aligned} fitness(i) = {} & w_1 \times \left(mean_i - mean_{target}\right) + \\ & + w_2 \times \left(variance_i - variance_{target}\right) + \\ & + w_3 \times \sum_{j=1}^{4} \left(contrast_i^j - contrast_{target}^j\right) + \\ & + w_4 \times \left(thickness_i - thickness_{target}\right) + \\ & + w_5 \times \left(interval_i - interval_{target}\right) + \\ & + w_6 \times \sum_{ce\ singularity\ Type} \left(singularity_i(c) + singularity_{target}(c)\right) + \\ & + w_7 \times \sum_{ce\ mintiae\ Type} \left(minutiae_i(c) + minutiae_{target}(c)\right) + \end{aligned} \tag{7}$$

The statistics of the target environment is calculated from the environment database and all the values are normalized from 0 to 1. The generated images (Figure 13) might be used to evaluate the performance of fingerprint recognition systems.

Figure 13. Generated fingerprints [31]

The usability of the proposed method was verified by comparing the fingerprints collected from real environments with those generated. Moreover, the proposed method has the applicability to the fingerprint image enhancement by modifying the fitness evaluation module.

3. Face Features

Face recognition is a biometric authentication method that has become more significant and relevant in recent years. It is becoming a more mature technology that has been employed in many large scale systems such as Visa Information System, surveillance access control and multimedia search engine [32]. Facial feature processing plays an important role in law enforcement forensic investigation [33], low bit video coding [34], security access control systems [35] and other applied and security systems. Generally, there are three categories of approaches for recognition, namely global facial feature, local facial feature and hybrid feature [32]. Systems using facial features can also be classified by source information used - 2D or 3D images and static images or video. Significant research has been done in EA application area for system using facial features improvement. Authors of [36] present an optimization approach that creates and successively improves Hausdorff Distance-Based Face Localization model by means of GA. To speed up the process and to prevent early saturation the researchers use a special bootstrapping method on the sample set and test several initialization functions. In [37] the high tolerance in human head movement and real-time processing that are needed for many applications, such as eye gaze tracking, is discussed. Template matching is used with GA, in order to overcome these problems. A high speed and accuracy tracking scheme using Evolutionary Video Processing for eye detection and tracking is proposed. Usually, a GA is unsuitable for a realtime processing, however, authors state that they have achieved real-time processing. The generality of this proposed method is provided by the artificial iris template used. In simulations the 97.9% eye tracking accuracy and an average processing time of 28 milliseconds per frame was achieved. Further we provide a detailed technical analysis of two research papers on GA application for 2D image and 3D image processing.

3.1 2D face image processing

An automatic facial feature extraction method is presented in [38]. The method is based on the edge density distribution of the image. In the preprocessing stage a face is approximated to an ellipse, and GA is applied to search for the best ellipse region match. In the feature extraction stage, GA is applied to extract the facial features, such as the eyes, nose and mouth, in the predefined sub regions. The normal process of searching for the features is computationally expensive, therefore GA is used as a search algorithm.

All the images used in experiments were head and shoulder images in a frontal view. Smoothing filters (median) were used for noise reduction. The face segmentation process was proceeded under the assumption that the face region can be approximated by an ellipsoid. This method works well even under the environments when the background is complex and the face contains extra features such as spectacles, beard and etc. The general processing scheme is presented on Figure 14.

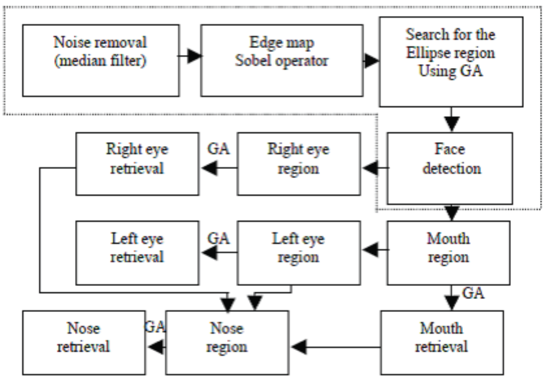

Figure 14. Block diagram of proposed feature extraction method [38]

Each chromosome in the population during the evolutionary search has five parameters genes, the center of the ellipse (x and y), x directional radius (r_x), y directional radius (r_y) and the angle (θ). The chromosome in binary form for each parameter is coded as shown in Figure 15.

x-8bits	y-8bits	r_x-8bits	r_y-8bits	θ-7bits

Figure 15. Chromosome for face segmentation [38]

The fitness function is defined by the number of edge pixels in the approximated ellipse like face to the actual number of pixels in the actual ellipse. The ratio is large when both ellipses overlap perfectly. It is commonly assumed in literature that the ratio of the length to breath of the face is 1.5: 1, therefore the same ratio is used to obtain the face area once the ellipse region is located. In the case of multiple faces in the image, faces are located until a threshold is satisfied. The threshold is based on the fitness value used to locate the faces. In the feature extraction stage, GA is used to search for the global maximum point when the template best matches the feature. The chromosome (Figure 16) represents the position of the feature in the x and y direction.

x-direction (7bits)	y-direction (7bits)

Figure 16. Chromosome for face feature extraction [38]

The fitness is evaluated in terms of the density of the template. The best template is selected when the fitness is maximized. The fitness function F is defined as:

$$F = \frac{1}{m \times n} \sum_{x=1}^{m} \sum_{y=1}^{n} T(x, y) \tag{8}$$

$$where \begin{cases} T(x, y) = 1 & if\ the\ pixel\ is\ white \\ T(x, y) = 0 & if\ the\ pixel\ is\ black \end{cases}$$

and T is the template, (x, y) are the coordinates of the template, and m, n is the size of the template.

Initially the population is chosen randomly. In each generation 20 percent of the population is considered for the reproduction. The Roulette Wheel selection scheme is applied in the selection process.

The proposed facial feature extraction approach has been validated with a large number of images. Some of the images contained more than one person, while others with person oriented at an angle. Simulation results showed that the facial features were extracted successfully. GA was able to search effectively and reduce computational complexity, therefore reduce the search time. The facial features were extracted even in the presence of artificial noise.

3.2 3D face image processing

Research on 3D face recognition has been intensified recently due to the significant advances of the 3D imaging technology. Most of the research focuses on the investigation of 3D range data obtained by a 3D scanner. Although 3D capture systems provide highly accurate 3D face information, it is not trivial to process the large amount of facial surface data. In model described in [39] each individualized facial model consists of 2953 vertices, 3D face model database is generated using 105 pairs of face images from 40 subjects. For each subject, there are two or three pairs of frontal and profile images, which were taken under different imaging conditions. In order to better characterize 3D features of the facial surface, each vertex on the individual model is labeled by one of eight label types. Therefore, the facial feature space is represented by a set of labels. A cubic approximation methods is explored to estimate the principal curvatures of each vertex on the model. Then the eight typical curvature types (i.e., convex peak, convex cylinder/cone, convex saddle, minimal surface, concave saddle, concave cylinder/cone, concave pit and planar) are categorized according to the relation of the principal curvatures. Figure 17a shows the labeled original feature space. Among the set of labels, only the labels located in certain regions are of the most interest.

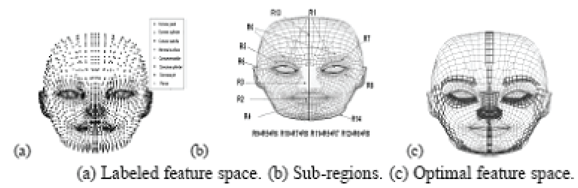

Figure 17. 3D facial feature extraction [39]

Some non-feature labels could be noises that may blur the individual facial characteristics. Therefore, a feature screening process to select features in order to better represent the individual facial traits for maximizing the difference between different subjects while minimizing the size of the feature space is applied. In order to select the optimal features, the face model is partitioned into 15 sub-regions based on their physical structures (there are overlaps between some of the regions, Figure 17b). Since not all the sub-regions contribute to the recognition task, and not all the vertices within one sub-region contribute to the classification, selection of the best set of vertex labels and the best set of sub-regions is needed. The purpose of the feature selection is to remove the irrelevant or redundant features which may degrade the performance of face classification. The GA is used successfully to address this type of problem.

The GA-based method selects the components that contribute the most to the face recognition task. The procedure for the GA-based feature selection consists of two parts:

1. vertices selection in each sub-region and
2. the integration of sub-regions.

In the first stage, equal error rate (EER) is used as the fitness function and those resulting in a higher EER rate are selected as good features. In the second stage, the sub-regions whose EER rate are higher than the mean EER rate value are integrated together as the final optimal feature spaces. Figure 17c shows the optimized feature space (dark color).

Two sets of databases were tested. One set consisted of 105 3D facial models and about 92% rank-four correct recognition rate was achieved. The other set had 387 models, the correct recognition rate was up to 87.6%. The experimental results showed that the features obtained from the 3D individualized model is feasible to classify and can be used to identify individual faces.

4. Other biometric information types and EA

In this chapter we provide a short overview of other biometric information processing techniques by the means of EA, which seam either promising for practical implementation in the nearest future (speaker recognition) or interesting from the scientific and novelty point of view (brain signals and activity style).

4.1 Speaker recognition

Speaker identification is one of the most important areas where biometric techniques can be used. Most published works in the areas of speech recognition and speaker recognition focus on speech under the noiseless environments and few published works focus on speech under noisy conditions. Learning systems in speaker identification that employ hybrid strategies can potentially offer significant advantages over single-strategy systems. In paper [40], Neuro-Genetic Hybrid algorithm with cepstral based features has been used to improve the performance of the text dependent speaker identification system under noisy environment.

At first algorithm performs the acquisition of speech utterances from speakers. To remove the background noises from the original speech, wiener filter is used. Then the detection algorithm is used to detect the start and end points from each speech utterance, after which the unnecessary parts are removed. Pre-emphasis filtering technique is used as a noise reduction technique to increase the amplitude of the input signal at frequencies where signal-to-noise ratio (SNR) is low. The speech signal is segmented into overlapping frames. The purpose of the overlapping analysis is that each speech sound of the input sequence would be approximately centered at some frame. After segmentation, windowing technique is used. Features are extracted from the segmented speech. The extracted features are then fed to the Neuro-Genetic hybrid techniques for learning and classification. Figure 18 shows the working process of neuro-genetic hybrid system.

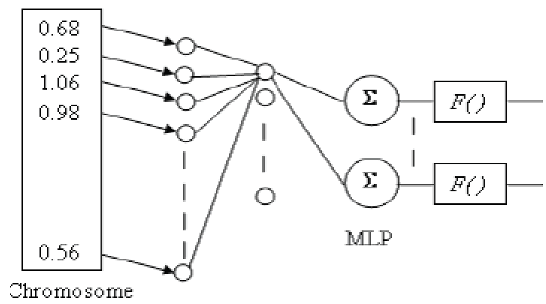

Figure 18. Learning and recognition model for the Neuro-Genetic hybrid system [40]

The structure of the multilayer neural network does not matter for the GA as long as the BPNs parameters are mapped correctly to the genes of the chromosome the GA is optimizing. Basically, each gene represents the value of a certain weight in the BPN and the chromosome is a vector that contains these values such that each weight corresponds to a fixed position in the vector as shown in Figure 18. The fitness function can be assigned from the identification error of the BPN for the set of frames used for training. The GA searches for parameter values that minimize the fitness function, thus the identification error of the BPN is reduced and the identification rate is maximized.

The experimental results has shown the versatility of the Neuro-Genetic hybrid algorithm based text-dependent speaker identification system. The critical parameters such as gain term, speed factor, number of hidden layer nodes, crossover rate and the number of generations have a great impact on the recognition performance of the proposed system. The optimum values of the above parameters have been selected effectively to find out the best performance. The highest recognition rate of BPN and GA have been achieved to be 94% and 95% respectively. According to VALID speech database, 100% identification rate in clean environment and 82.33% in office environment conditions have been achieved in Neuro-Genetic hybrid system. Therefore, this proposed system can be used in various security and access control purposes.

4.2 Activity style identification and brain signals processing

An online biometric verification system for use over the Internet that requires no specific equipment was presented in [41]. Combining two distinct tests to ensure authenticity, a typing style test and a mouse-based signature test, the fraudulent access rate was equal to 4.4%, while authentic users could access the system with a rate of 99 %.

Signal recording from the brain is rather complicated biometrics, based on brain signals. It has not been studied extensively yet though it is one of the most fraud resistant biometrics. It is unlikely that different persons will have similar activity in all parts of the brain. The research article [42] describes the electroencephalogram (EEG) based method which uses features computed from 61 Visual Evoked Potential (VEP) signals and states that VEP signals are the most suitable for identification of individuals. Fischer Discriminant Ratio (FDR) has been used to find the optimal EEG channels to reduce the computational time. However, the fusion of GA with Linear Discriminant Analysis (LDA) classifier shows that the identification performance is improved compared to FDR.

5. Conclusions

In this article the biometry definition was provided, the main and perspective biometry types were listed, the general overview of evolutionary algorithms, such as genetic algorithm and genetic programming, was provided, and evolutionary algorithm application review for biometric system reliability and quality improvement was provided.

The review has shown, that evolutionary algorithm application in the specified area is perspective and may insure qualitative increase of biometric system parameters, such as speed, error rate and flexibility. The biggest research currently done is in sphere of fingerprint and facial feature processing, several works on other biometric information types, such as voice and brain signals, also exist.

On the other hand the review allowed determining that most of the proposed systems are of a prototype level and production tests are necessary to be certain in their reliability and suitability for practical application. One more perspective research area is application of methods proposed for one biometric information type for another. It is also expected that evolutionary algorithm usage will increase with the increase of requirements to biometric systems and technologic progress in currently technologically complex biometric systems.

References

1. J. Abbazio, S. Perez, D. Silva, R. Tesoriero, F. Penna and R. Zack, Proc. Student-Faculty Research Day, CSIS, New York, USA, pp. C1.1-C1.8 (2009).
2. A. Jain, R. Bole and S. Pankanti, BIOMETRICS: Personal Identification in Networked Society, Kluwer Academic Press, Boston (1999).
3. J. D. Woodward, Biometrics: A Look at Facial Recognition, RAND Corporation (2003).
4. U. Uludag, S. Pankanti, S. Prabhakar and A. K. Jain, Proc. of the IEEE, 92, 948 (2004).
5. S. Pankanti, S. Bolle and A. K. Jain, IEEE Computer (Special Issue on Biometrics), 33, 46 (2000).
6. D. Hurley, M. Nixon and J. Carter, Computer Vision and Image Understanding, 98, 491 (2005).
7. L. Biel, O. Pettersson, L. Philipson and P. Wide, IEEE Transactions on Instrument and Measurement, 50, 808 (2001).
8. K. V. R. Ravi and R. Palaniappan, Soft Computing, 10, 163 (2006).
9. http://www.sans.org/infosecFaQ/authentic/face_rec.htm
10. D. E. Goldberg, Genetic Algorithms in Search, Optimization,and Machine Learning, Addison-Wesley (1989).
11. J. Huang and H.Wechsler, Proc. of 2nd International Conference on Audio and Video-Based Biometric Person Authentication (AVBPA), Washington, DC, USA, 8pp. (1999).
12. J. Holland, Adoption in natural and artificial systems, The MIT press (1975).
13. K. S. Tang, K. F. Man, S. Kwong and Q. Hem, IEEE Signal Processing Magazine, 13, 22 (1996).
14. N. Goranin, A. Cenys, Information Technology And Control, 37, 133 (2008).
15. I. A. Ismail, N. A. ElRamly, M. A. Abd-ElWahid, P. M. ElKafrawy and M. M. Nasef, International Journal of Computer Science and Information Security, 6, 316 (2009).
16. http://www.genetic-programming.org
17. J. R. Koza, Proc. of WESCON Conference, Piscataway, NJ, USA, pp.589-594 (1995).
18. I. M. M. El-Emary and M. M. Abd El-Kareem, World Applied Sciences Journal, 5, 276 (2008).
19. http://iris.sel.eesc.usp.br/lavi/pdf/D__poster_22526.pdf
20. S. Pankanti, S. Prabhakar, A. Jain, IEEE Trans., 24, 1010 (2002).
21. A. K. Jain, L. Hong, S. Pankanti and R. Bolle, IEEE Trans., 85, 1365 (1997).
22. D. Maio, and D. Maltoni, IEEE Trans. Pattern Analysis Machine Intelligence, 19, 27 (1997).
23. S. Prabhakar, A. K. Jain and S. Pankanti, Pattern recognition, 36, 1847 (2003).
24. http://www.uh.edu/engines/fingerprint_minutiae.jpg
25. http://www.emeraldinsight.com/fig/0870210403001.png
26. X. Tan and B. Bhanu, Proc. of IEEE Workshop on Applications of Computer Vision, Orlando, Florida, USA, pp. 79-79 (2002).
27. T. Scheidat, A. Engel and C. Vielhauer, Proc. of the 8th workshop on Multimedia and Security, Geneva, Switzerland, pp.130-134 (2006).
28. H. R. Sudarshana Reddy and N. V. Subba Reddy, Lecture Notes in Computer Science, 3285, 9 (2004).

29. M. R. Girgisa, A. A. Sewisyb and R. F. Mansourc, GVIP Journal, 7, 51 (2007).
30. D. Maltoni, Automatic Fingerprint Recognition Systems, N. Ratha (eds.), R. Bolle (eds.), Springer, Heidelberg (2004).
31. U. K. Cho, J. H. Hong and S. B. Cho, LNAI, 4570, 444 (2007).
32. D. Mohamad, International Journal of Image Processing, 1, 1 (2009).
33. R. Brunelli and T. Poggio IEEE Transactions on Pattern Analysis and Machine Intelligence, 15, 1042 (1993).
34. M. Chuang, R. Chang and L. Huang, Journal of Information Science and Engineering, 16, 447 (2000).
35. C. Lin and W. Ling, IEEE Transactions on Image Processing, 8, 834 (1999).
36. K. J. Kirchberg, O. Jesorsky and R. W. Frischholz, Lecture Notes in Computer Science, 2359, 103 (2002).
37. T. Akashi, Y. Wakasa and K. Tanaka, Systemics, Cybernetics and Informatics, 5, 72 (2007).
38. G. G. Yen and N. Nithianandan, Proc. of Proceedings of the 2002 Congress of the Evolutionary Computation CEC '02, pp.1895-1900 (2002).
39. Y. Sun and L. Yin, Computational Imaging and Vision, 35, 95 (2007).
40. R. Islam, and F. Rahman, International Journal of Computer Science Issues, 1, 42 (2009).
41. R. A. J. Everitt and P. W. McOwan, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25, 1166 (2003).
42. http://fit.mmu.edu.my/news/newsattach/EXTENDED%20ABSTRACT_1.1_Ravi.pdf