

New Method for Dynamic Signature Verification Based on Global Features

Marcin Zalasinski¹, Krzysztof Cpalka¹, and Yoichi Hayashi²

¹ Częstochowa University of Technology,
Institute of Computational Intelligence, Poland
{marcin.zalasinski,krzysztof.cpalka}@iisi.pcz.pl

² Meiji University, Department of Computer Science, Japan
hayashiy@cs.meiji.ac.jp

Abstract. Identity verification based on the dynamic signatures is commonly known issue of biometrics. This process is usually done using methods belonging to one of three approaches: global approach, local function based approach and regional function based approach. In this paper we focus on global features based approach which uses the so called global features extracted from the signatures. We present a new method of global features selection, which are used in the training and classification phase in a context of an individual. Proposed method bases on the evolutionary algorithm. Moreover, in the classification phase we propose a flexible neuro-fuzzy classifier of the Mamdani type. Our method was tested using the SVC2004 public on-line signature database.

1 Introduction

Signature is a biometric attribute which is commonly used in the process of identity verification. It belongs to the group of behavioural attributes, like gait (see e.g. [17]), related to the characteristic of individual's behaviour. Verification based on these attributes is more difficult than verification based on the physiological ones, like face or iris (see e.g. [1], [41]-[43], [64]), but it is less invasive.

Signature biometric attribute may be classified into two categories - static (off-line) signature and dynamic (on-line) signature. Static signature, which contains only information about shape of the trajectory, is more common in everyday life (it is on many paper documents), but identity verification based on this type of signature is less reliable than verification based on the dynamic signature. Dynamic signature contains also information about dynamics of the signing process, e.g. velocity, acceleration and pressure. Shape of the on-line signature is represented by the horizontal and vertical trajectories. Methods of the dynamic signature verification may be categorized into three main groups (see e.g. [11]): global features based methods, local function based methods and regional function based methods. Global features based methods use so called global features which are extracted from the signature and used during training and classification phase (see e.g. [32], [36], [67]). Examples of these features are signature total

duration and number of pen-ups. Function based methods compare time functions, which contains information about changes of signature features over time (see e.g. [18], [24]-[25]). In this approach waveforms extracted from the signature are compared to the waveforms of the other signature and classification is made on the basis of this process result. Regional based methods rely on segmentation of signature into some regions, used during training and verification phase (see e.g. [66]-[69]).

In this paper we focus on the approach based on global features. We use a set of global features proposed in [19], which contains extended collection of features from three other papers - [31], [37]-[38]. It should be noted that the operation of our method is not dependent on the adopted feature set, which can be practically arbitrarily reduced or extended. In the approach proposed in this paper, large global feature set is reduced by selection of optimal features subset, which is considered during classification phase. Moreover, global features are ranked and only features with the highest rank value are used in the classification process. Application of evolutionary feature selection in the proposed algorithm is possible thanks to using a new fuzzy one-class classifier.

The problem of global features selection has been considered in the literature (see e.g. [19], [31]). Please note that the method proposed in this paper stands out from the methods of other authors by following characteristics: **(a)** The proposed method takes advantage of an evolutionary algorithm in the process of feature selection. **(b)** The proposed method uses in the classification process a hierarchy of features individually for each user. **(c)** The proposed method takes advantage of the theory of fuzzy sets and fuzzy systems (see e.g. [2]-[6], [12]-[15], [28])

This paper is organized into four sections. In Section 2 we present idea of the new method for dynamic signature verification based on global features. In Section 3 simulation results are presented. Conclusions are drawn in Section 4.

2 Idea of the New Method for Dynamic Signature Verification Based on Global Features

Idea of the proposed method can be summarized as follows: **(a)** It works on the basis of a set of 85 features describing the dynamics of the dynamic signature which have been systematized, for example, in the paper [19]. As already mentioned, the proposed method does not depend on the base set of features. This set can be freely modified. **(b)** It uses an evolutionary algorithm with specifically defined evaluation function. The function promotes chromosomes (which correspond to the solutions) encoding a set of features, whose values are homogeneous within the training signatures of the user. **(c)** It uses (developed for the considered method) one-class classifier which is based on the capacities of the flexible fuzzy system proposed by us earlier (see e.g. [11], [69]). It allows to take into account the weights of importance of individual features, selected individually for each user. **(d)** It works in two modes: (1) learning and (2) testing (operating mode). In the first mode the selection of features is performed for each user, descriptors of features and weights of importance of features are

determined. They are needed for proper work of the classifier in the test phase. These parameters are stored in a database. In the second mode, mode of operation (verification of test signature), the parameters stored for each user in the learning phase are downloaded from the database and then signature verification is realized on the basis of these parameters. It should be noted that the efficiency of the method does not depend on the number of users whose signatures are stored in a database (thus descriptors heterogeneity of individual characteristics of different users is not taken into account). This encumbrance has been introduced intentionally, because it causes that the effectiveness of the method in practical applications does not depend on the number of records in the database. Of course, in the learning phase (as already mentioned) skilled forgeries are not used (they are only used for test of the method), which is an additional advantage of the proposed approach.

2.1 Training Phase

General description of the training phase for the user i (procedure **Training**(i)) can be described as follows: **Step 1.** Acquisition of J training signatures of user i . **Step 2.** Determination of the matrix \mathbf{G}_i of all considered global features, describing dynamics of signatures, for all J training signatures of the user i . **Step 3.** Determination of the vector $\bar{\mathbf{g}}_i$ of average values for each global feature, determined in **Step 2** for J training signatures of the user i . **Step 4.** Evolutionary selection of subset of global features, which are the most characteristic for the user i (procedure **EvolutionaryFeaturesSelection**($\mathbf{G}_i, \bar{\mathbf{g}}_i$)). This process is performed on the basis of a similarity of features values computed using chosen distance measure. **Step 5.** Determination of the vector \mathbf{X}'_i which contains information about selected global features characteristic for the user i . Length of the vector (denoted as N) is equal to the number of global features selected in **Step 4**. Please note that the vector \mathbf{X}'_i is in practice the best chromosome from the population considered in the **Step 4** ($\mathbf{X}'_i = \mathbf{X}_{i, chBest}$). **Step 6.** Selection of classifier parameters used in the test phase (procedure **ClassifierDetermination**($i, \mathbf{X}'_i, \mathbf{G}_i, \bar{\mathbf{g}}_i$)). **Step 7.** Storing in a database the following information about the user i : vector \mathbf{X}'_i , vector $\bar{\mathbf{g}}_i$, parameters of classifier $maxd_{i,n}$ and $w_{i,n}$ ($n = 1, \dots, N$).

Later in this section steps of the procedure **Training**(i) have been described in details.

First, acquisition of training signatures for the user i is performed (**Step 1**). Next, the matrix \mathbf{G}_i , which contains all considered global features of all J training signatures of user i , is determined (**Step 2**):

$$\mathbf{G}_i = \begin{bmatrix} g_{i,1,1} & g_{i,2,1} & \dots & g_{i,N,1} \\ g_{i,1,2} & g_{i,2,2} & \dots & g_{i,N,2} \\ \vdots & \vdots & \ddots & \vdots \\ g_{i,1,J} & g_{i,2,J} & \dots & g_{i,N,J} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{i,1} \\ \mathbf{g}_{i,2} \\ \vdots \\ \mathbf{g}_{i,N} \end{bmatrix}, \tag{1}$$

where $\mathbf{g}_{i,j} = [g_{i,1,j} \ g_{i,2,j} \ \dots \ g_{i,N,j}]$, $g_{i,n,j}$ is a value of the global feature, $i = 1, 2, \dots, I$ is an index of the user, I is a number of the users, $n = 1, 2, \dots, N$ is a number of the global feature, $j = 1, 2, \dots, J$ is an index of the signature, J is a number of the signatures created by the user in the acquisition phase (which is a part of the training phase).

In the **Step 3** vector $\bar{\mathbf{g}}_i$ of average values of each global feature of all training signatures J of user i is determined:

$$\bar{\mathbf{g}}_i = [\bar{g}_{i,1}, \bar{g}_{i,2}, \dots, \bar{g}_{i,N}], \quad (2)$$

where $\bar{g}_{i,n}$ is average value of the global feature n of training signatures of the user i , computed using the following formula (**Step 3**):

$$\bar{g}_{i,n} = \frac{1}{J} \sum_{j=1}^J g_{i,n,j}. \quad (3)$$

In the next step (**Step 4**) selection of the optimal subset of global features for the user i is performed. For this purpose the algorithm **EvolutionaryFeaturesSelection**($\mathbf{G}_i, \bar{\mathbf{g}}_i$), described in Section 2.2, is used. The procedure **EvolutionaryFeaturesSelection**($\mathbf{G}_i, \bar{\mathbf{g}}_i$) returns the subset of global features \mathbf{X}'_i which are used during creation of the classifier (**Step 5**). In the **Step 6** classifier for the user i is determined. Next, all data required in the process of classifier determination and signature verification (vector $\mathbf{X}_{i,ch}$, vector $\bar{\mathbf{g}}_i$, parameters of the classifier $maxd_{i,n}$ and $w_{i,n}$) are stored into a database (**Step 7**). Detailed description of the classifier determination is presented in Section 2.4.

2.2 Evolutionary Features Selection

The main step in the learning phase **Training**(i) is **Step 4**, in which evolutionary selection of features for the user i is performed. The procedure of features selection is called **EvolutionaryFeaturesSelection**($\mathbf{G}_i, \bar{\mathbf{g}}_i$). Remarks on the considered procedure can be summarized as follows: **(a)** It uses the binary encoding in which each of the genes of individual chromosomes encodes the information whether the corresponding feature has to be taken into account in the process of signature verification of considered user i (a gene encoding a value "1" means that the feature associated with this gene has to be considered in the process of signature verification). Thus, each of the chromosomes has a length corresponding to the number of all considered features and encodes a subset of the features. Further in the paper, it is assumed that $\mathbf{X}_{i,ch} = [X_{i,ch,g=1}, X_{i,ch,g=2}, \dots, X_{i,ch,g=N}]$ means chromosome with index ch , $ch = 1, 2, \dots, Ch$, in a population associated with the user i , whose number of genes is equal to the number of features (the value N). **(b)** The procedure **EvolutionaryFeaturesSelection**($\mathbf{G}_i, \bar{\mathbf{g}}_i$) is consistent with the typical scheme of the evolutionary algorithm, therefore it will not be considered in detail. It includes the initialization of the population, population evaluation, selection of chromosomes from the population, the evolution of the chromosomes in the population carried by the application of evolutionary operators (in the simulations

we use crossover and mutation), checking of the stopping criterion. A detailed description of the algorithm can be found, among others, in [8], [67]. (c) The originality of the proposed approach results from a specific way of determining the evaluation function of chromosomes from the population. Evaluation of the chromosomes is based on the similarity of features for the user's reference signatures created in the training phase. The objective of the algorithm is to minimize the evaluation function, thus such features are preferred, whose values (determined for the reference signatures created in the training phase) for the user are the most similar to each other according to the adopted measure of similarity. Details of the procedure $\text{CalculateFf}(i, \mathbf{G}_i, \bar{\mathbf{g}}_i, \mathbf{X}_{i,ch})$ are described in the Section 2.3. (d) Evolutionary features selection can be performed using other algorithms based on the population, which differ in their approach to exploration and exploitation of a space of considerations. (e) The result of the procedure $\text{EvolutionaryFeaturesSelection}(\mathbf{G}_i, \bar{\mathbf{g}}_i)$ is the information about the set of features describing the stability of signing in the learning phase by the user i . This information is stored in the best chromosome of the last step of performed evolution process. Next, it is rewritten into a vector \mathbf{X}'_i , whose length is equal to N (as a length of the chromosome $\mathbf{X}_{i,ch}$).

2.3 Determination of Fitness Function

In the definition of the fitness function of the chromosome, the following input parameters are taken into account: (a) i - an index of the user for which the training process is performed. (b) \mathbf{G}_i - a matrix of all global features values, determined for all reference signatures of the user i . (c) $\bar{\mathbf{g}}_i$ - a vector of average values of global features, averaged in the context of all reference signatures of the user i . (d) $\mathbf{X}_{i,ch}$ - a chromosome with index ch in the population associated with the user i , for which the value of the evaluation function is calculated.

General description of the procedure for determining the evaluation function of chromosomes belonging to the population and encoding subsets of features ($\text{CalculateFf}(i, \mathbf{G}_i, \bar{\mathbf{g}}_i, \mathbf{X}_{i,ch})$) is the following: **Step 1.** Determination of the covariance matrix for the matrix of all global features. It should be noted that during determination of the covariance matrix only the global features from the subset of the features encoded in the chromosome $\mathbf{X}_{i,ch}$ are taken into account. In the further description of the method, the matrix of the subset of global features created by combining \mathbf{G}_i and $\mathbf{X}_{i,ch}$ will be denoted as \mathbf{G}' and the covariance matrix corresponding to the matrix \mathbf{G}' will be denoted as $\text{cov}(\mathbf{G}')$. A vector containing the elements of the row j of the matrix \mathbf{G}' will be denoted as \mathbf{g}'_j . Number of rows of the matrix \mathbf{G}' results from the number of reference signatures of the user i created during acquisition phase (training) and it is equal to J . Number of columns of the matrix \mathbf{G}' results from the number of features encoded in the chromosome $\mathbf{X}_{i,ch}$ (the number of non-zero elements in the vector). Please note that the matrix \mathbf{G}' will not be used anywhere outside the procedure $\text{CalculateFf}(i, \mathbf{G}_i, \bar{\mathbf{g}}_i, \mathbf{X}_{i,ch})$. **Step 2.** Determination of the vector

of Mahalanobis distances (see e.g. [16]) \mathbf{m} between the vector of average values of the global features and the matrix of the global features values. It should be noted that only the global features from the subset of features encoded in the chromosome $\mathbf{X}_{i,ch}$ are taken into account during determination of the Mahalanobis distances vector \mathbf{m} . Thus, in the further description of the method the vector of average values of the subset of global features created by combining $\bar{\mathbf{g}}_i$ and $\mathbf{X}_{i,ch}$ will be denoted as $\bar{\mathbf{g}}'$. It will be used during determination of the value of the vector \mathbf{m} . Number of elements of the vector \mathbf{m} results from the number of reference signatures of the user i created in the acquisition phase (training) and it is equal to J . Please note that vectors $\bar{\mathbf{g}}'$ and \mathbf{m} will not be used anywhere outside the procedure $\text{CalculateFf}(i, \mathbf{G}_i, \bar{\mathbf{g}}_i, \mathbf{X}_{i,ch})$. **Step 3.** Determination of the evaluation function of the chromosome $\mathbf{X}_{i,ch}$. Value of this function (denoted as $\text{ff}(\mathbf{X}_{i,ch})$) is determined by averaging the values of the Mahalanobis distances vector \mathbf{m} .

Later in this section a detailed description of the function $\text{CalculateFf}(i, \mathbf{G}_i, \bar{\mathbf{g}}_i, \mathbf{X}_{i,ch})$ is provided.

In the **Step 1** covariance matrix $\text{cov}(\mathbf{G}')$ of global features encoded in the chromosome is created. Covariance is a measure of the linear correlation between the global features of the reference signatures of the user (and created in the acquisition phase). Thus, the covariance matrix $\text{cov}(\mathbf{G}')$ is a square matrix $N \times N$, where N is a number of features. Next, values m_j , $j = 1, 2, \dots, J$, of the vector of Mahalanobis distances are determined using the following formula:

$$m_j = \sqrt{(\mathbf{g}' - \bar{\mathbf{g}}')^T (\text{cov}(\mathbf{G}'))^{-1} (\mathbf{g}' - \bar{\mathbf{g}}')}. \quad (4)$$

It should be noted that for each subset of features J distances are created. The subset of features associated with the lowest distance is the most valuable for the user i in the training phase. In the **Step 3** value of the fitness function of the chromosome $\mathbf{X}_{i,ch}$ is determined as follows:

$$\text{ff}(\mathbf{X}_{i,ch}) = \frac{1}{J} \sum_{j=1}^J m_j. \quad (5)$$

Lower value of the fitness function means that the chromosome is "better" (subset of global features encoded in the chromosome $\mathbf{X}_{i,ch}$ is the most characteristic for the user i).

2.4 Determination of Classifier

General form of the procedure $\text{ClassifierDetermination}(i, \mathbf{X}'_i, \mathbf{G}_i, \bar{\mathbf{g}}_i)$, which determines parameters of the our classifier, can be presented as follows:

Step 1. Determination of Euclidean distances $d_{i,n,j}$ between each global feature n encoded in the chromosome \mathbf{X}'_i and average value of the global feature for all J signatures of the user i . **Step 2.** Selection of maximum distance for each global feature n from distances determined in **Step 1**. It should be emphasized that the maximum distance (labelled as $\text{max}d_{i,n}$) are individual for each user i .

They will be used in the classification phase of the signature (verification of the authenticity). Therefore, they must be stored in a database (in addition to the parameters: vector \mathbf{X}'_i , vector $\bar{\mathbf{g}}_i$). **Step 3.** Computation of weights of importance $w_{i,n}$, associated with the feature number n of the user i and used in the classification phase. It should be emphasized that the weights also have individual character for the user i and they will be used in the classification process of the signature. Therefore, they must be stored in a database. **Step 4.** Creation of the flexible neuro-fuzzy system using values determined in **Step 2** and **Step 3**.

In the **Step 1** distances $d_{i,n,j}$ between each global feature n encoded in \mathbf{X}'_i and average value of the global feature for all J signatures of the user i is computed using the following formula:

$$d_{i,n,j} = X''_{i,n} \cdot \sqrt{(\bar{\mathbf{g}}_{i,n} - \mathbf{g}_{i,n,j})^2}, \tag{6}$$

where $X'_{i,n} \in \{0, 1\}$ is gene value of the chromosome \mathbf{X}'_i , associated with the feature number n . Next, maximum distance for each global feature is selected (**Step 2**):

$$\max d_{i,n} = \max_{j=1, \dots, J} \{d_{i,n,j}\}. \tag{7}$$

In the **Step 3** weights of importance of features $w_{i,n}$ for each global feature n of the user i are determined. Weight of the global feature n of the user i is computed on the basis of standard deviation of the global feature n of the user i and average value of distances for the global feature n of the user i (computed in the **Step 2**). This process is described by the following formula:

$$w_{i,n} = \frac{\sqrt{\frac{1}{J} \sum_{j=1}^J (g_{i,n} - g_{i,n,j})^2}}{\frac{1}{J} \sum_{j=1}^J d_{i,n,j}}. \tag{8}$$

Next, a classifier is created (**Step 4**). We use flexible neuro-fuzzy system of the Mamdani type (see e.g. [50]). Neuro-fuzzy systems (see e.g. [27], [56], [60]-[63]) combine the natural language description of fuzzy systems (see e.g. [21]-[23], [29]-[30], [46]-[48]) and the learning properties of neural networks (see e.g. [7], [26], [33]-[34], [39]-[40], [44]-[45], [53]-[55], [57]-[58], [66]). Alternative approaches to classification can be found in [49], [51]-[52]. Our system is based on the rules in the form if-then. The fuzzy rules contain fuzzy sets which represent the values, e.g. "low" and "high", of the input and output linguistic variables. In our method the input linguistic variables are dependent on the similarity between the global features of the test signature and average values of global features computed on the basis of training signatures. The system uses only features selected individually for the user during evolutionary selection process. Output linguistic variables describe the reliability of the signature. In our method input

parameters of fuzzy sets are individually selected for each user (**Step 2** of the procedure `ClassifierDetermination`($i, \mathbf{X}'_i, \mathbf{G}_i, \bar{\mathbf{g}}_i$)). Please note that if training signatures are more similar to each other, the tolerance of our classifier is lower. The flexibility of the classifier results from the possibility of using in the classification the importance of global features, which are selected individually for each user (**Step 3** of the procedure `ClassifierDetermination`($i, \mathbf{X}'_i, \mathbf{G}_i, \bar{\mathbf{g}}_i$)). Taking into account the weights of importance of the global features is possible thanks to the use of proposed by us earlier (see e.g. [20]) aggregation operators named the weighted triangular norms.

Our system works on the basis of two fuzzy rules presented as follows:

$$\left\{ \begin{array}{l} R^{(1)} : \left[\text{IF } (dst_{i,1} \text{ is } A_{i,1}^1) \mid w_{i,1} \text{ AND } (dst_{i,2} \text{ is } A_{i,2}^1) \mid w_{i,2} \text{ AND } \dots \right. \\ \quad \left. \text{AND } (dst_{i,N} \text{ is } A_{i,N}^1) \mid w_{i,N} \text{ THEN } y_i \text{ is } B^1 \right] \\ R^{(2)} : \left[\text{IF } (dst_{i,1} \text{ is } A_{i,1}^2) \mid w_{i,1} \text{ AND } (dst_{i,2} \text{ is } A_{i,2}^2) \mid w_{i,2} \text{ AND } \dots \right. \\ \quad \left. \text{AND } (dst_{i,N} \text{ is } A_{i,N}^2) \mid w_{i,N} \text{ THEN } y_i \text{ is } B^2 \right] \end{array} \right\}, \quad (9)$$

where **(a)** $A_{i,n}^1, A_{i,n}^2, i = 1, 2, \dots, I, n = 1, 2, \dots, N$, are input fuzzy sets related to the global feature number n of the user i . Fuzzy sets $A_{i,1}^1, A_{i,2}^1, \dots, A_{i,N}^1$ represent values "high" assumed by input linguistic variables $dst_{i,n}$ in the test phase and variables $d_{i,n,j}$ in the training phase, computed using (6), both for signatures in the training phase and the test phase. Analogously, fuzzy sets $A_{i,1}^2, A_{i,2}^2, \dots, A_{i,N}^2$ represent values "low" assumed by input linguistic variables $dst_{i,n}$ in the test phase and variables $d_{i,n,j}$ in the training phase. Thus, each rule contains N antecedents. In the fuzzy classifier of the signature used in the simulations we applied a Gaussian membership function (see Fig. 1) for all input fuzzy sets. **(b)** $y_i, i = 1, 2, \dots, I$, is output linguistic variable interpreted as reliability of the signature considered to be created by the signer i . **(c)** B^1, B^2 are output fuzzy sets shown in Fig. 1. Fuzzy set B^1 represents value "high" of output linguistic variable determining the reliability of the signature. Analogously, fuzzy set B^2 represents value "low" of output linguistic variable determining the reliability of the signature. In the fuzzy classifier of the signature used in the simulations we applied the membership function of type γ in the rule 1 and the membership function of type L in the rule 2. Please note that the membership function of fuzzy sets B^1 and B^2 are the same for all users. **(d)** $w_{i,n}, i = 1, 2, \dots, I, n = 1, 2, \dots, N$, are weights of importance related to the global feature number n of the user i .

2.5 Identity Verification Phase

The process of signature verification (`SignatureVerification`(i)) is performed in the following way: **Step 1.** Acquisition of test signature of the user which is considered as user i . **Step 2.** Download of information about selected features of the user i (\mathbf{X}'_i), average values of this features computed during training phase ($\bar{\mathbf{g}}_i$) and classifier parameters of the user i from the database ($maxd_{i,n}, w_{i,n}$). **Step 3.** Determination of values of global features which have been selected as the most characteristic for the user i in the training phase. **Step 4.** Verification of the test signature using one class flexible neuro-fuzzy classifier.

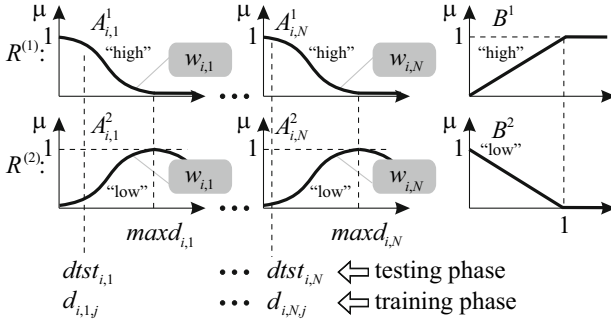


Fig. 1. Input and output fuzzy sets of the flexible neuro-fuzzy system of the Mamdani type for verification signature of user i

In the **Step 1** user which identity will be verified creates one test signature. In this step user claims his identity as i . Next, information about selected features of the user i (\mathbf{X}'_i), average values of this features computed during training phase ($\bar{\mathbf{g}}_i$) and parameters of the classifier of the user i created during training phase ($maxd_{i,n}, w_{i,n}$) are downloaded from the database (**Step 2**). In the **Step 3** system determines global features of the test signature. Finally, verification is performed using flexible one-class neuro-fuzzy classifier of the Mamdani type (**Step 4**). A signature is true if the following assumption is satisfied:

$$\bar{y}_i = \frac{T^* \left\{ \mu_{A^1_{i,1}}(dtst_{i,1}), \dots, \mu_{A^1_{i,N}}(dtst_{i,N}); w_{i,1}, \dots, w_{i,N} \right\}}{\left(T^* \left\{ \mu_{A^1_{i,1}}(dtst_{i,1}), \dots, \mu_{A^1_{i,N}}(dtst_{i,N}); w_{i,1}, \dots, w_{i,N} \right\} + T^* \left\{ \mu_{A^2_{i,1}}(dtst_{i,1}), \dots, \mu_{A^2_{i,N}}(dtst_{i,N}); w_{i,1}, \dots, w_{i,N} \right\} \right)} > cth_i, \tag{10}$$

where **(a)** $T^* \{ \cdot \}$ is the algebraic weighted t-norm (see [50]), **(b)** $\mu_A(\cdot)$ is a Gaussian membership function, **(c)** $\mu_{B^1}(\cdot)$ is a membership function of the class L , **(d)** $\mu_{B^2}(\cdot)$ is a membership function of the class γ , **(e)** $\bar{y}_i, i = 1, 2, \dots, I$, is the value of the output signal of applied neuro-fuzzy system described by rules (9). Detailed description of the system can be found in [9], [20], **(e)** Formula (10) was created by taking into account in the description of system simplification resulting from the spacing of fuzzy sets, shown in Fig. 1. The simplifications can be described as follows: $\mu_{A^1_{i,n}}(0) = 1, \mu_{A^1_{i,n}}(maxd_{i,n}) \approx 0, \mu_{A^2_{i,n}}(0) \approx 0, \mu_{A^2_{i,n}}(maxd_{i,n}^{\{s\}}) = 1$. Detailed information about the system described by the rules in the form (9), which allow to easily derive the relationship (10) on the basis of the above assumptions, can be found e.g. in [9]-[10], [20], **(f)** $cth_i \in [0, 1]$ - coefficient determined experimentally for each user to eliminate disproportion between FAR and FRR error (see e.g. [65]).

3 Simulation Results

Simulations were performed using SVC 2004 public database (see [65]). During the simulations the following assumptions have been adopted: **(a)** population contains 100 chromosomes, **(b)** algorithm stops after the lapse of a determined number of 1000 generations, **(c)** during selection of chromosomes tournament selection method is used, **(d)** crossover is performed with probability equal to 0.8 at three points, **(e)** mutation is performed for each gene with probability equal to 0.02. Details concerning the interpretation of these parameters can be found, among others, in [50], [59].

The database contains 40 signers and for each signer 20 genuine and 20 forgery signatures. The test was performed five times, every time for all signers stored in the database. In training phase 5 genuine signatures (numbers 1-10) of the signer were used. During test phase 10 genuine signatures (numbers 11-20) and 20 forgery signatures (numbers 21-40) of each signer were used. Simulations were performed in the authorial environment implemented in C#.

During simulation we tested three methods of verification based on global features. The one of them was our method described in this paper. Results of the simulations are presented in the Table 1. The table contains values of FAR (False Acceptance Rate) and FRR (False Rejection Rate) errors which are commonly used in the literature to evaluate the effectiveness of identity verification methods (see e.g. [18], [25]).

Moreover, in Fig. 2 information on the frequency of selection of individual features are presented. Each sample in the graph refers to the global feature of the signature and it is a percentage value of the frequency of the feature selection in the context of the 40 users.

Table 1. Results of simulation performed by our system

Method	Average FAR	Average FRR	Average error
PCA using random subspace [35]	25.75 %	24.60 %	25.18 %
Evolutionary selection with PCA [67]	23.87 %	22.65 %	23.26 %
Our method	16.69 %	13.18 %	14.94 %

Conclusions of the simulations can be summarized as follows: **(a)** The accuracy of our method is higher in comparison to the methods described in [35] and [67]. The method proposed in this paper works with clearly greater accuracy for considered database SVC 2004. **(b)** Before carrying out the simulation we expected that in the process of evolutionary selection of features some of them may be chosen more often than others in the context of all users. However, it turned out that none of the features had not dominated the others. This may indicate that the most important in the context of identity verification of the user are sets of features (combinations of features), not their individual features. In addition, it can be seen that the five features have never been selected in the process of evolution.

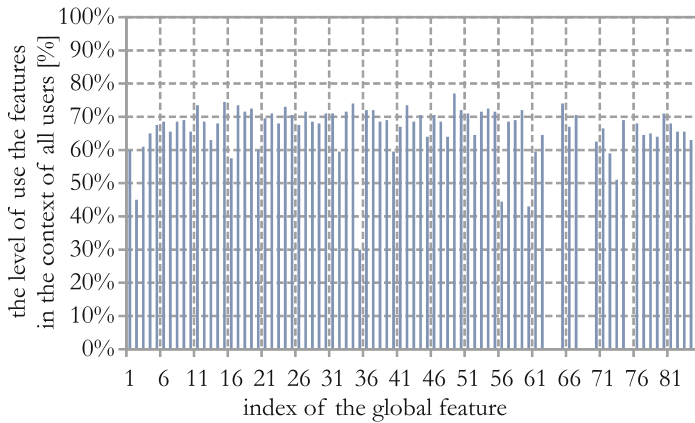


Fig. 2. Percentage frequency of selection of the global features of the signature for users from the database SVC2004

4 Conclusions

In this paper a new method for dynamic signature verification based on global features is presented. The method assumes selection of the subset of global features from a large set of the features. This process is performed using evolutionary algorithm. Its application was made possible by a well-defined evaluation function, which takes into account the diversity of values of the reference signatures for each user and does not require any signatures of other users. The features are selected individually for each user. It is worth noting that during the selection of features none of them clearly dominates the other, what may indicate that the most characteristic for the user are individual combinations of features, not individual features. The achieved accuracy of the signature verification in comparison with other methods proves correctness of the assumptions.

Acknowledgment. The project was financed by the National Science Centre (Poland) on the basis of the decision number DEC-2012/05/B/ST7/02138.

References

1. Abiyev, R.H., Altunkaya, K.: Neural network based biometric personal identification with fast iris segmentation. *International Journal of Control, Automation and Systems* 7, 17–23 (2009)
2. Bartzuk, Ł., Dziwiński, P., Starczewski, J.T.: A new method for dealing with unbalanced linguistic term set. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part I. LNCS*, vol. 7267, pp. 207–212. Springer, Heidelberg (2012)
3. Bartzuk, Ł., Dziwiński, P., Starczewski, J.T.: New Method for Generation Type-2 Fuzzy Partition for FDT. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part I. LNCS*, vol. 6113, pp. 275–280. Springer, Heidelberg (2010)

4. Bartczuk, L., Przybył, A., Dziwiński, P.: Hybrid state variables - fuzzy logic modelling of nonlinear objects. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part I. LNCS, vol. 7894, pp. 227–234. Springer, Heidelberg (2013)
5. Bartczuk, L., Rutkowska, D.: A New Version of the Fuzzy-ID3 Algorithm. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS (LNAI), vol. 4029, pp. 1060–1070. Springer, Heidelberg (2006)
6. Bartczuk, L., Rutkowska, D.: Medical Diagnosis with Type-2 Fuzzy Decision Trees. In: Kącki, E., Rudnicki, M., Stempczyńska, J. (eds.) Computers in Medical Activity. AISC, vol. 65, pp. 11–21. Springer, Heidelberg (2009)
7. Bilski, J., Smola, J.: Parallel Approach to Learning of the Recurrent Jordan Neural Network. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part I. LNCS, vol. 7894, pp. 32–40. Springer, Heidelberg (2013)
8. Cpałka K., Łapa K., Przybył A., Zalasinski M.: A new method for designing neuro-fuzzy systems for nonlinear modelling with interpretability aspects, *Neurocomputing* (in press, 2014), <http://dx.doi.org/10.1016/j.neucom.12.031>
9. Cpałka, K., Rutkowski, L.: Flexible Takagi Sugeno Neuro-fuzzy Structures for Non-linear Approximation. *WSEAS Transactions on Systems* 9(4), 1450–1458 (2005)
10. Cpałka, K.: A Method for Designing Flexible Neuro-fuzzy Systems. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS (LNAI), vol. 4029, pp. 212–219. Springer, Heidelberg (2006)
11. Cpałka, K., Zalasinski, M.: On-line signature verification using vertical signature partitioning. *Expert Systems with Applications* 41, 4170–4180 (2014)
12. Dziwiński, P., Bartczuk, L., Starczewski, J.T.: Fully controllable ant colony system for text data clustering. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) EC 2012 and SIDE 2012. LNCS, vol. 7269, pp. 199–205. Springer, Heidelberg (2012)
13. Dziwiński, P., Rutkowska, D.: Algorithm for generating fuzzy rules for WWW document classification. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS (LNAI), vol. 4029, pp. 1111–1119. Springer, Heidelberg (2006)
14. Dziwiński, P., Rutkowska, D.: Ant focused crawling algorithm. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2008. LNCS (LNAI), vol. 5097, pp. 1018–1028. Springer, Heidelberg (2008)
15. Dziwiński, P., Starczewski, J.T., Bartczuk, L.: New linguistic hedges in construction of interval type-2 FLS. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010, Part II. LNCS, vol. 6114, pp. 445–450. Springer, Heidelberg (2010)
16. De Maesschalck, R., Jouan-Rimbaud, D., Massart, D.L.: The Mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems* 50, 1–18 (2000)
17. Ekinci, M., Aykut, M.: Human Gait Recognition Based on Kernel PCA Using Projections. *Journal of Computer Science and Technology* 22, 867–876 (2007)
18. Faundez-Zanuy, M.: On-line signature recognition based on VQ-DTW. *Pattern Recognition* 40, 981–992 (2007)
19. Fierrez-Aguilar, J., Nanni, L., Lopez-Peñalba, J., Ortega-Garcia, J., Maltoni, D.: An On-Line Signature Verification System Based on Fusion of Local and Global Information. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 523–532. Springer, Heidelberg (2005)
20. Gabryel, M.: Cpałka K., Rutkowski L, Evolutionary strategies for learning of neuro-fuzzy systems. In: Proceedings of the I Workshop on Genetic Fuzzy Systems, Granada, pp. 119–123 (2005)

21. Greenfield, S., Chiclana, F.: Type-reduction of the discretized interval type-2 fuzzy set: approaching the continuous case through progressively finer discretization. *Journal of Artificial Intelligence and Soft Computing Research* 1(3), 183–193 (2011)
22. Horzyk, A., Tadeusiewicz, R.: Self-Optimizing Neural Networks. In: Yin, F.-L., Wang, J., Guo, C. (eds.) *ISSN 2004. LNCS*, vol. 3173, pp. 150–155. Springer, Heidelberg (2004)
23. Jelonkiewicz, J., Przybył, A.: Accuracy improvement of neural network state variable estimator in induction motor drive. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 71–77. Springer, Heidelberg (2008)
24. Jeong, Y.S., Jeong, M.K., Omिताomu, O.A.: Weighted dynamic time warping for time series classification. *Pattern Recognition* 44, 2231–2240 (2011)
25. Kholmatov, A., Yanikoglu, B.: Identity authentication using improved online signature verification method. *Pattern Recognition Letters* 26, 2400–2408 (2005)
26. Korytkowski, M., Nowicki, R., Rutkowski, L., Scherer, R.: AdaBoost Ensemble of DCOG Rough–Neuro–Fuzzy Systems. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) *ICCCI 2011, Part I. LNCS*, vol. 6922, pp. 62–71. Springer, Heidelberg (2011)
27. Korytkowski, M., Rutkowski, L., Scherer, R.: On combining backpropagation with boosting. In: *Proceedings of the IEEE International Joint Conference on Neural Network (IJCNN)*, vol. 1-10, pp. 1274–1277 (2006)
28. Korytkowski, M., Rutkowski, L., Scherer, R.: From Ensemble of Fuzzy Classifiers to Single Fuzzy Rule Base Classifier. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 265–272. Springer, Heidelberg (2008)
29. Kroll, A.: On choosing the fuzziness parameter for identifying TS models with multidimensional membership functions. *Journal of Artificial Intelligence and Soft Computing Research* 1(4), 283–300 (2011)
30. Li, X., Er, M.J., Lim, B.S., Zhou, J.H., Gan, O.P., Rutkowski, L.: Fuzzy Regression Modeling for Tool Performance Prediction and Degradation Detection. *International Journal of Neural Systems* 20(5), 405–419 (2010)
31. Lee, L.L., Berger, T., Aviczer, E.: Reliable on-line human signature verification systems. *IEEE Trans. on Pattern Anal. and Machine Intell.* 18, 643–647 (1996)
32. Lumini, A., Nanni, L.: Ensemble of on-line signature matchers based on overcomplete feature generation. *Expert Systems with Applications* 36, 5291–5296 (2009)
33. Łapa, K., Przybył, A., Cpałka, K.: A new approach to designing interpretable models of dynamic systems. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II. LNCS*, vol. 7895, pp. 523–534. Springer, Heidelberg (2013)
34. Łapa, K., Zalaśiński, M., Cpałka, K.: A new method for designing and complexity reduction of neuro-fuzzy systems for nonlinear modelling. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part I. LNCS*, vol. 7894, pp. 329–344. Springer, Heidelberg (2013)
35. Nanni, L.: Experimental comparison of one-class classifiers for online signature verification. *Neurocomputing* 69, 869–873 (2006)
36. Nanni, L., Lumini, A.: Ensemble of Parzen window classifiers for on-line signature verification. *Neurocomputing* 68, 217–224 (2005)
37. Nelson, W., Kishon, E.: Use of dynamic features for signature verification. In: *Proc. of the IEEE Intl. Conf. on Systems, Man, and Cyber*, vol. 1, pp. 201–205 (1991)
38. Nelson, W., Turin, W., Hastie, T.: Statistical methods for on-line signature verification. *Intl. Journal of Pattern Recognition and Artificial Intell.* 8, 749–770 (1994)

39. Nowicki, R.: Rough-Neuro-Fuzzy System with MICOG Defuzzification. In: 2006 IEEE International Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence, Vancouver, BC, Canada, July 16-21, pp. 1958–1965 (2006)
40. Nowicki, R., Scherer, R., Rutkowski, L.: A method for learning of hierarchical fuzzy systems. In: Sincak, P., Vascak, J., Kvasnicka, V., Pospichal, J. (eds.) *Intelligent Technologies - Theory and Applications*, pp. 124–129. IOS Press (2002)
41. Pabiasz, S., Starczewski, T.J.: Face reconstruction for 3D systems. In: *Selected Topics in Computer Science Applications*, pp. 54–63. EXIT (2011)
42. Pabiasz, S., Starczewski, J.T.: Meshes vs. depth maps in face recognition systems. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part I. LNCS*, vol. 7267, pp. 567–573. Springer, Heidelberg (2012)
43. Pabiasz, S., Starczewski, J.T.: A New Approach to Determine Three-Dimensional Facial Landmarks. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II. LNCS*, vol. 7895, pp. 286–296. Springer, Heidelberg (2013)
44. Patan, K., Patan, M.: Optimal Training strategies for locally recurrent neural networks. *Journal of Artificial Intelligence and Soft Computing Research* 1(2), 103–114 (2011)
45. Peteiro-Barral, D., Bardinias, B.G., Perez-Sanchez, B.: Learning from heterogeneously distributed data sets using artificial neural networks and genetic algorithms. *Journal of Artificial Intelligence and Soft Computing Research* 2(1), 5–20 (2012)
46. Pławiak P., Tadeusiewicz R, Approximation of phenol concentration using novel hybrid computational intelligence methods. *Applied Mathematics and Computer Science* 24(1) (in print, 2014)
47. Przybył, A., Jelonekiewicz, J.: Genetic algorithm for observer parameters tuning in sensorless induction motor drive. In: Rutkowski, L., Kacprzyk, J. (eds.) *Neural Networks And Soft Computing* (6th International Conference on Neural Networks and Soft Computing), Zakopane, Poland, pp. 376–381 (2003)
48. Przybył, A., Smolağ, J., Kimla, P.: Distributed Control System Based on Real Time Ethernet for Computer Numerical Controlled Machine Tool (in Polish). *Przegląd Elektrotechniczny* 86(2), 342–346 (2010)
49. Rutkowski, L.: An application of multiple Fourier series to identification of multi-variable nonstationary systems. *International Journal of Systems Science* 20(10), 1993–2002 (1989)
50. Rutkowski, L.: *Computational intelligence*. Springer (2008)
51. Rutkowski, L.: Nonparametric learning algorithms in the time-varying environments. *Signal Processing* 18, 129–137 (1989)
52. Rutkowski, L.: The real-time identification of time-varying systems by nonparametric algorithms based on the Parzen kernels. *International Journal of Systems Science* 16, 1123–1130 (1985)
53. Rutkowski, L.: Flexible structures of neuro-fuzzy systems. In: Sincak, P., Vascak, J. (eds.) *Quo Vadis Computational Intelligence. SCI*, vol. 54, pp. 479–484. Springer, Heidelberg (2000)
54. Rutkowski, L., Cpałka, K.: Flexible weighted neuro-fuzzy systems. In: *Proceedings of the 9th International Conference on Neural Information Processing (ICONIP 2002)*, Orchid Country Club, Singapore, November 18-22 (2002)
55. Rutkowski, L., Przybył, A., Cpałka, K., Er, M.J.: Online speed profile generation for industrial machine tool based on neuro-fuzzy approach. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part II. LNCS*, vol. 6114, pp. 645–650. Springer, Heidelberg (2010)

56. Rutkowski, L., Przybył, A., Cpałka, K.: Novel on-line speed profile generation for industrial machine tool based on flexible neuro-fuzzy approximation. *IEEE Transactions on Industrial Electronics* 59, 1238–1247 (2012)
57. Scherer, R.: Neuro-fuzzy relational systems for nonlinear approximation and prediction. *Nonlinear Analysis Series A: Theory, Methods and Applications* 71(12), e1420–e1425 (2009)
58. Scherer, R., Rutkowski, L.: Connectionist fuzzy relational systems. In: Halgamuge, S.K., Wang, L. (eds.) 9th International Conference on Neural Information and Processing; 4th Asia-Pacific Conference on Simulated Evolution and Learning; 1st International Conference on Fuzzy Systems and Knowledge Discovery, Singapore. *Computational Intelligence for Modelling and Prediction. SCI*, vol. 2, pp. 35–47. Springer, Heidelberg (2005)
59. Sivanandam, S.N., Deepa, S.N.: *Introduction to Genetic Algorithms*. Springer (2008)
60. Starczewski, J.T.: A Type-1 Approximation of Interval Type-2 FLS. In: Di Gesù, V., Pal, S.K., Petrosino, A. (eds.) *WILF 2009. LNCS*, vol. 5571, pp. 287–294. Springer, Heidelberg (2009)
61. Starczewski, J.T., Rutkowski, L.: Connectionist Structures of Type 2 Fuzzy Inference Systems. In: Wyrzykowski, R., Dongarra, J., Paprzycki, M., Waśniewski, J. (eds.) *PPAM 2001. LNCS*, vol. 2328, pp. 634–642. Springer, Heidelberg (2002)
62. Starczewski, J., Rutkowski, L.: Interval type 2 neuro-fuzzy systems based on interval consequents. In: Rutkowski, L., Kacprzyk, J. (eds.) *Neural Networks and Soft Computing. Advances in Soft Computing*, pp. 570–577. Springer, Heidelberg (2003)
63. Starczewski, J.T., Scherer, R., Korytkowski, M., Nowicki, R.: Modular Type-2 Neuro-fuzzy Systems. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Wasniewski, J. (eds.) *PPAM 2007. LNCS*, vol. 4967, pp. 570–578. Springer, Heidelberg (2008)
64. Xu, G., Zhang, Z., Ma, Y.: A novel method for iris feature extraction based on intersecting cortical model network. *Journal of Applied Mathematics and Computing* 26, 341–352 (2008)
65. Yeung, D.Y., Chang, H., Xiong, Y., George, S., Kashi, R., Matsumoto, T., Rigoll, G.: *SVC2004: First International Signature Verification Competition*. In: Zhang, D., Jain, A.K. (eds.) *ICBA 2004. LNCS*, vol. 3072, pp. 16–22. Springer, Heidelberg (2004)
66. Zalasinski, M.: Cpałka K, A new method of on-line signature verification using a flexible fuzzy one-class classifier, pp. 38–53. Academic Publishing House EXIT (2011)
67. Zalasinski, M., Łapa, K., Cpałka, K.: New algorithm for evolutionary selection of the dynamic signature global features. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II. LNCS*, vol. 7895, pp. 113–121. Springer, Heidelberg (2013)
68. Zalasinski, M., Cpałka, K.: Novel algorithm for the on-line signature verification. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part II. LNCS*, vol. 7268, pp. 362–367. Springer, Heidelberg (2012)
69. Zalasinski, M., Cpałka, K.: New approach for the on-line signature verification based on method of horizontal partitioning. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II. LNCS*, vol. 7895, pp. 342–350. Springer, Heidelberg (2013)