Scientific Research Publishing

Journal of Modern Physics, 2018, 9, 112-129
http://www.scirp.org/journal/jmp
ISSN Online: 2153-120X
ISSN Print: 2153-1196

Keystroke Dynamics Based Authentication Using Possibilistic Renyi Entropy Features and Composite Fuzzy Classifier

Aparna Bhatia[1*], Madasu Hanmandlu[2]

[1]Department of Electrical Engineering, Indian Institute of Technology Delhi, HauzKhas, New Delhi, India
[2]CSE Department, MVSR Engg. College, Nadergul, Hyderabad, Formally with EE Department, IIT Delhi, New Delhi
Email: *aparna.bhatia@gmail.com

How to cite this paper: Bhatia, A. and Hanmandlu, M. (2018) Keystroke Dynamics Based Authentication Using Possibilistic Renyi Entropy Features and Composite Fuzzy Classifier. *Journal of Modern Physics*, **9**, 112-129.
https://doi.org/10.4236/jmp.2018.92008

Received: November 18, 2017
Accepted: January 16, 2018
Published: January 19, 2018

Abstract

This paper presents the formulation of the possibilistic Renyi entropy function from the Renyi entropy function using the framework of Hanman-Anirban entropy function. The new entropy function is used to derive the information set features from keystroke dynamics for the authentication of users. A new composite fuzzy classifier is also proposed based on Mamta-Hanman entropy function and applied on the Information Set based features. A comparison of the results of the proposed approach with those of Support Vector Machine and Random Forest classifier shows that the new classifier outperforms the other two.

Keywords

Keystroke Dynamics, Information Set, Renyi Entropy Function and Its Possibilistic Version, Composite Fuzzy Classifier

1. Introduction

With ever increasing use of computers, internet and online transactions, the need for access control and web security is necessitated. Various methods are used for secure access and online user authentication but most of them are afflicted with some drawbacks. Password/PIN based access control for user authentication is easy to be forged using brute force attack. Similarly, tokens like smart cards used for authentication get lost or easily stolen. So, biometric systems employing both physiological and behavioral modalities have recently gained popularity. The physiological biometrics comprising physical traits such as face, fingerprint, iris, palm-print, speech and hand geometry has gained pop-

ularity in recent years. Behavioral biometrics is based on the human physical activity like gait, voice, signature and keystroke dynamics. Keystroke Dynamics depicts a natural typing rhythm captured through keyboard available on most computing systems including hand held devices like mobile/PDAs which possess touch based keyboard. Keystroke Dynamics is a strong behavioral biometrics with many advantages and offers solution to many problems faced with other access control mechanisms. Some of the advantages include: It cannot be copied as it is difficult to copy the human behavior, and it cannot be stolen, forged or lost. As no special device is required, it is a low-cost biometrics solution. Keystroke Dynamics has high user acceptance and can be operated in hidden mode. It can also be used for continuous user authentication while user is working on the system. Moreover, keystroke dynamics based authentication is best suited for online user verification as keystroke features comprise not so large timing vector.

1.1. Literature Survey

The features for keystroke dynamics mainly consist of timing data like time to move from one key to another also known as flight time and time for which a key is pressed also known as dwell time. Different researchers have used different timing features based on the above basic keystroke timings. Some researchers have used Down-Down Time, Up-Up Time, Up-Down and Down-Up Time where Down Time is the time instance when key is pressed and Up Time is the time instance when key is released. Similarly, Press-Press Time (PPTime), Press-Release Time (PRTime), Release-Press Time (RPTime) and Release-Release Time (RRTime) are used. Press Time is the same as Down Time and Release Time is the same as Up Time. In addition to timing features we can also include keystroke pressure, *i.e.* the pressure applied on the key, as part of keystroke dynamics features [1] [2], but this pressure measurement requires a special hardware; hence used scarcely.

The text entry in the form of fixed string predetermined at the initial instance of user interaction with the authentication system for extracting keystroke dynamics is static. Text entry can also be dynamic where user types the free text for continuous authentication of a user. The static entry datasets are publicly available in Killorhy and Maxion [3], Giot *et al.* [4], Loy *et al.* [1] [2]. The free text databases are:Biochaves by Filho and Freire [5] and Clarksons University Keystroke Dataset by Vural *et al.* [6].

User's master profile is created based on keystroke dynamics behavior from username using trajectory dissimilarity technique in [7]. Master trajectory profile of the user is created by averaging the trajectories of the first 10 input records and used as authentication mechanism in addition to the user's password. By this method the best results of 4% equal error rate or 96% authentication accuracy are achieved.

Killourhy and Maxion [3] have collected keystroke data of 51 users with 400

samples for each user and evaluated 14 detectors on the collected data. The error rates and the dataset collected are shared publicly to establish a benchmark for comparison. Hosseinzadeh and Krishnan [8] have used UUKL (Up-Up Keystroke latency) feature and made a comparison with other keystroke features using GMM based verification system. In this UUKL outperforms commonly used hold time and down-down keystroke latency. The user specific adaptive threshold is based on leave-one-out method (LOOM) that achieves EER of 4.4% on the dataset of 41 users. Çeker and Upadhyaya [9] have also used GMM with dynamic text of 30 users and obtained Equal Error Rate (EER) of 0.08% with 2 components whereas pure Gaussian gives EER of 1.3% for continuous authentication based system.

Deng and Zhong [10] have used GMM-UBM (Gaussian Mixture Model with Universal Background Model) and DBN (Deep Belief Networks) wherein the data from background users is employed as imposters' data during the training phase. The EERs of 0.055 and 0.035 are achieved with GMM-UBM and DBN respectively.

Teh *et al.* [11] have proposed a statistical fusion approach using Gaussian Probability Density function and Direction Similarity Measure (DSM) which evaluates the consistency of user's typing behavior. DSM is the difference in signs between the two consecutive keystrokes in a phrase. By this approach the best EER of 6.36% is obtained with the weighted sum rule on their own dataset.

A hybrid model involving the fusion of Gaussian probability density function (GPDF) and SVM based scores is developed in [12]. The mean and standard deviation are calculated from the training feature vectors that serve as template during testing. The scores are then calculated using GPDF and SVM and the score-level fusion is applied using four fusion rules. Best results are achieved with the combination of Press-Release and Release-Press time-measurements using the weighted sum rule.

Pisani *et al.* [13] have used the enhanced template update which adapts the user model as per the changes in the typing behavior over time. The templates are updated by considering the negative samples, *i.e.* samples classified as imposters in addition to the genuine samples. The experimental results show better predictive performance in terms of the reduced FMR (False Match Rate) and FNMR (False Non- Match Rate).

Ivannikova *et al.* [14] have introduced dependence clustering based approach for user authentication using keystroke dynamics. Cross validation process is designed and artificially generated impostor samples are used to improve the learning process. The best results in terms of EER of 0.077 and ZMFAR of 0.358 are achieved on CMU benchmark dataset due to Dependence Clustering using Manhattan distance.

Sliding windows of different sizes are used in [15] for template update methods. The double threshold method employs two thresholds: One update threshold to decide if query can be used for reference template update and another verification threshold to decide if a query is accepted or denied. It is shown that

user-specific threshold that varies from one session to another because of the update mechanism yields lower error rates than those with the fixed threshold.

1.2. Motivation

The present work is concerned with the generation of information set features using the possibilistic Renyi entropy function from the keystroke dynamics comprising dwell time and flight time. Our previous work [16] deals with generation of the information set features from the same measurements of keystroke dynamics but uses the Mamta-Hanman entropy function in [17].

Though many classifiers falling under statistical methods, neural networks and pattern recognition techniques are in vogue for the authentication of a user using keystroke dynamics, we propose a new fuzzy classifier.

The organization of the paper is as follows: Section 2 gives the derivation for the possibilistic Renyi Entropy function. It also formulates the Information Set features and higher form of these features based on this entropy function. Section 3 develops an algorithm for the Composite Fuzzy classifier based on Composite convex Entropy function. Section 4 describes the databases used in the present work and Section 5 discusses the results of implementation. Section 6 gives the conclusions.

2. Renyi Entropy

To represent the probabilistic uncertainty, we have several entropy functions such as Shannon, Pal and Pal [18], Renyi [19], and Hanman-Anirban entropy functions [20]. Of which Hanman-Anirban entropy being an information theoretic entropy function is capable of representing both probabilistic and possibilistic uncertainties. In this work, we would like to investigate the suitability of Renyi entropy function for representing the possibilistic uncertainty because it has one free parameter which we can cash in to meet our objective. The original Renyi Entropy function is given by:

$$H_R = \frac{1}{1-\alpha}\log\left(\sum_{i=1}^{n} p_i^{\alpha}\right) \tag{1}$$

To represent the possibilistic uncertainty, p_i is replaced by T_i in (1). This leads to

$$H_R = \frac{1}{1-\alpha}\log\left(\sum_{i=1}^{n} T_i^{\alpha}\right) \tag{2}$$

The unknown parameter in (2) is constant but we take it as a variable in the range (0, 1) and derive in the next section the adaptive Renyi entropy function by relating it to the Hanman-Anirban entropy function [20] given by

$$H_i = T_i \mathrm{e}^{-\left[aT_i^3+bT_i^2+cT_i+d\right]} \tag{3}$$

where T_i is the information source value and a, b, c, and d are the parameters in the exponential gain function. These parameters are selected to be statistical parameters such that this gain function becomes the Gaussian function. For this

the choice of parameters is: $a=0$, $b=\frac{1}{2\sigma^2}$, $c=-\frac{2\bar{T}}{2\sigma^2}$ and $d=\frac{\bar{T}^2}{2\sigma^2}$ where $\bar{T}$ is the mean value. Then (3) becomes $H_i=T_i\mu_i$.

2.1. Adaptive Renyi Function

To bring (2) into the possibilistic domain, let us consider only i[th] term in the summation and α to be a variable. This leads to

$$H_i=\frac{1}{1-\alpha_i}\log\left(T_i^{\alpha_i}\right)=\frac{\alpha_i}{1-\alpha_i}\log\left(T_i\right) \tag{4}$$

Assuming the membership function value as $\mu_i=\frac{1}{1-\alpha_i}$, we have $\frac{\alpha_i}{1-\alpha_i}=-\bar{\mu}_i$. The membership function μ is taken to be Gaussian function with its statistical parameters, the mean $\bar{T}$ and the variance σ^2 computed from the keystroke measurements $\{T\}$ as explained above.

Now replacing this in (4) we have i[th] component of adaptive Renyi entropy function is:

$$H_i=-\bar{\mu}_i\log T_i \tag{5}$$

This r.h.s. of this equation is represented in modus ponen form as: $\bar{\mu}_i\to\log T_i$. This in turn allows us to write $-\bar{\mu}_i\cup\log T_i$, which means we can write $H_i=\max\{-\bar{\mu}_i,\log T_i\}$, though we have taken it as the product.

Replacing $-\log p=\mathrm{e}^{1-p}$ in (5), we get one term of the adaptive Renyi entropy function as:

$$H_i=\bar{\mu}_i\mathrm{e}^{(1-T_i)} \tag{6}$$

This is different from the entropy function term, $T_i^{\alpha}\mu_i^{\beta}$ derived in our previous work [16]. Looking at these two terms we notice that T_i and μ_i assume opposite roles. When T_i is an information source value, μ_i acts as a gain function value. Their roles are interchanged in (6), *i.e.* the complement membership function acts as the information source value and T_i appears in the gain function. Thus the relations between information source value and the gain function value are shown to be varied and such different forms help us try on different applications. Recall the one term of Pal and Pal entropy function in [18], that is $T_i\mathrm{e}^{(1-T_i)}$ and comparing this with (6) we find that only the information value differs. As $\bar{\mu}_i$ is a function of T_i involving statistical parameters it will have more flexibility if it is chosen as Gaussian function.

The above is in the form given by

$$H_i=f\left(T_i\right)\cdot\mathrm{e}^{(1-T_i)} \tag{7}$$

This is an information value in Hanman-Anirban entropy function for $a=b=0$, $c=1$, $d=-1$ and replacing T_i by a function of T as $f\left(T_i\right)=\bar{\mu}_i$. Thus the information source value is a complement membership function $\bar{\mu}_i$ and the gain function is exponential. We have shown that one term of Renyi

function takes one specific form of the Hanman-Anirban entropy function. From (6) it is easy to form an information set: $\left\{\overline{\mu}_i \mathrm{e}^{(1-T_i)}\right\}$ by varying the index i and the resulting possibilistic Renyi entropy function is:

$$H_{Rp} = \sum_{i=1}^{n} \overline{\mu}_i \mathrm{e}^{(1-T_i)} \tag{8}$$

Let the mean membership be $\mu = \frac{1}{n}\sum_{i=1}^{n} \mu_i$ and substituting this in (1) for α, we have

$$H_R = \frac{1}{1-\mu} \log\left(\sum_{i=1}^{n} T_i^{\mu}\right) \tag{9}$$

The difference $\Delta H_R = H_R - H_{Rp}$ is the error incurred in the approximation of Renyi function in the possibilistic domain.

2.2. Some Functions of Adaptive Renyi Entropy Function

1) *Complement Renyi Function*: By replacing $\overline{\mu}$ with μ in Equation (6) we get Complement Renyi function as:

$$H_i = \mu_i \mathrm{e}^{(1-T_i)} \tag{10}$$

The above can be written as $H_i = \mu_i\left(1-1+T_i\right) = \mu_i T_i$. With the substitution of proper values for the parameters in (2), we get what we call the basic information value $\mu_i T_i$. This is proved in [16] [21] [22] [23]; so Equation (10) is a variant of this entropy function.

2) *Sigmoid Renyi Function*: Considering Equation (6) as a unit of information, we will now apply a sigmoid function on it to get:

$$S_i = \frac{1}{1+\mathrm{e}^{-\overline{\mu}_i \mathrm{e}^{(1-T_i)}}} \tag{11}$$

3) *Complement Sigmoid Function*: Replacing μ with $\overline{\mu}$ in (8), we get:

$$S_i = \frac{1}{1+\mathrm{e}^{-\mu_i \mathrm{e}^{(1-T_i)}}} \tag{12}$$

4) *Renyi Entropy Energy*: This follows from (8) by multiplying it with $\overline{\mu}$.

$$H_i = \overline{\mu}_i^2 \mathrm{e}^{(1-T_i)} \tag{13}$$

5) *Complement Renyi Energy*: By taking complement of $\overline{\mu}$, we obtain this as:

$$H_i = \mu_i^2 \mathrm{e}^{(1-T_i)} \tag{14}$$

6) *Renyi Transform*: Renyi entropy function is not amenable for conversion to transforms just as Hanman transform. When we put Renyi entropy function into the form of Hanman-Anirban entropy function, it offers us the facility to create transforms. Consider the Hanman-Anirban entropy function in the following form:

$$H_i = f\left(T_i\right) \mathrm{e}^{-\left[aT_i^3 + bT_i^2 + cT_i + d\right]} \tag{15}$$

where $f\left(T_i\right) = T_i$ in the original Hanman-Anirban entropy function. But we

take $\overline{\mu}_i = f(T_i)$ to convert into the Renyi entropy function form. Further taking $a=0$, $b=0$, $c=\mu_i$ and $d=0$ we get the Renyi transform given by:

$$H_i = \overline{\mu}_i \mathrm{e}^{-\mu_i T_i} \tag{16}$$

To introduce non-linearity in the values of $\overline{\mu}_i$ we can modify it as a power of α

$$H_i = \overline{\mu}_i^{\alpha} \mathrm{e}^{-\mu_i T_i} \tag{17}$$

7) *Complement Renyi Transform*: Taking complement $\overline{\mu}_i$ in place of μ_i we get Complement Renyi Transform as:

$$H_i = \mu_i \mathrm{e}^{-\mu_i T_i} \tag{18}$$

8) *Modified Sigmoid Renyi Function*: Applying sigmoid function to Equation (5), we get Modified Sigmoid Renyi Function as:

$$S_i = \frac{1}{1+\mathrm{e}^{-\overline{\mu}_i \log T_i}} \tag{19}$$

9) *Modified Complement Sigmoid Renyi Function*: Taking complement in Equation (19) by replacing μ with $\overline{\mu}$ we get the modified complement sigmoid Renyi function as:

$$S_i = \frac{1}{1+\mathrm{e}^{-\mu_i \log T_i}} \tag{20}$$

2.3. The Two-Component Information Set (TCIS)

In our previous work [16] we have proposed the use of Two Component Information System (TCIS) features for Keystroke Dynamics and the results were promising. In this approach, the first component I_1 represents the temporal information and the membership function μ_1 is derived from the data that includes all the training feature vectors. The second component I_2 represents the spatial information and the membership function μ_2 is derived using all the features contained in a single sample. When the above two information components are concatenated, Two-Component Information set features are obtained denoted by *I*. The concatenated features are input to the classifier for authentication.

Algorithm [16]:

Step 1: Calculate mean $\left(T_{avg}^{(1)}\right)$ and variance $\left(\sigma^{(1)}\right)$ of all the training samples.

Step 2: Calculate mean $\left(T_{avg}^{(2)}\right)$ and variance $\left(\sigma^{(2)}\right)$ of all the keystroke features in a single training sample.

Step 3: Compute $\mu^{(1)}$ using $T_{avg}^{(1)}$ and $\sigma^{(1)}$ and similarly compute $\mu^{(2)}$ using $T_{avg}^{(2)}$ and $\sigma^{(2)}$. Next compute two components, $I_1 = \left\{\mu_{ij}^{(1)} T_{ij}\right\}$ and $I_2 = \left\{\mu_{ij}^{(2)} T_{ij}\right\}$ using $\mu^{(1)}$ and $\mu^{(2)}$.

Step 4: Concatenate I_1 and I_2 to form *I*. Then train any classifier using concatenated *I*.

Step 5: Compute I_{t1} using $T_{avg}^{(1)}$ and $\sigma^{(1)}$ from Step 1 for each test sample.

Step 6: Compute mean $\left(T_{avg}^{(2)}\right)$ and variance $\left(\sigma^{(2)}\right)$ of all the features in the test sample. Also compute I_{t2} using $T_{avg}^{(2)}$ and $\sigma^{(2)}$.

Step 7: Concatenate I_{t1} and I_{t2} to obtain I_t and feed this feature vector to any classifier.

3. Composite Fuzzy Classifier

Design of a Composite Fuzzy Classifier

Before proceeding to the design of a classifier we need the error vector between the training feature vector of l^{th} user corresponding to m^{th} training sample denoted by x_{mj}^{l} and the test feature vector t_j. Let the size of the feature vector be n and the number of training feature vectors be s for each user. The error vector is computed from:

$$e_{mj}^{l} = \left| x_{mj}^{l} - t_j \right|;\ m = 1,2,\cdots,s,\ j = 1,2,\cdots,n \tag{21}$$

As we need a membership in the formulation of a fuzzy classifier, we select an exponential membership function as:

$$\mu_{mj}^{l} = \mathrm{e}^{-\left(\left|x_{mj}^{l} - t_j\right|\right)};\ m = 1,2,\cdots,s,\ j = 1,2,\cdots,n \tag{22}$$

In view of (21), Equation (22) is rewritten as

$$\mu_{mj}^{l} = \mathrm{e}^{-e_{mj}^{l}} \tag{23}$$

We now apply Frank t-norm (t_F) on a pair of error vectors e_{mj}^{l} and e_{hj}^{l} to yield the normed error vector denoted by $E_{mh}^{l}(k)$ as follows:

$$E_{mh}^{l}(j) = t_F\left(e_{mj}^{l}, e_{hj}^{l}\right);\ m \neq h,\ j = 1,2,\cdots,n \tag{24}$$

In the above, t_F is given by

$$t_F = \log_q\left[1 + \frac{\left(q^{e_{mj}^{l}} - 1\right)\left(q^{e_{hj}^{l}} - 1\right)}{q - 1}\right];\ k = 1,2,\cdots,V \tag{25}$$

Similarly, we compute t-norm of a pair of membership functions μ_{mj}^{l} and μ_{hj}^{l} called the normed membership function using:

$$M_{mh}^{l}(j) = t_F\left(\mu_{mj}^{l}, \mu_{hj}^{l}\right),\ m \neq h,\ j = 1,2,\cdots,n \tag{26}$$

As proved in [23] that the information value is the product of information source value and the corresponding membership function value. Considering $E_{mh}^{l}(j)$ as the information source vector and $M_{mh}^{l}(j)$ as the corresponding membership function vector, their product $\left\{E_{mh}^{l}(j) M_{mh}^{l}(j)\right\}$ gives the information vector.

Derivation of Composite Entropy Function: For this derivation, we take recourse to Mamta-Hanman entropy function in the form:

$$H = \sum_{j=1}^{n} T_j^{\alpha} \mathrm{e}^{-\left(cT_j^{\gamma} + d\right)^{\beta}} \tag{27}$$

By substituting $c = -1$, $d = 0$ and $\beta = 1$ we obtain

$$H = \sum_{j=1}^{n} T_j^{\alpha} \mathrm{e}^{T_j^{\gamma}} \tag{28}$$

with $T_j = E_{mh}^{l}(j) M_{mh}^{l}(j)$. To develop the composite entropy function, we apply logarithmic function on (28) leading to

$$H = \log \sum_{j=1}^{n} T_j^{\alpha} \mathrm{e}^{T_j^{\gamma}} \tag{29}$$

The composite function is the result of applying logarithmic function on Mamta-Hanman entropy function. That is, we are modifying the entropy value by the logarithmic function. In this case the available information is Mamta-Hanman entropy value which we are modifying by applying logarithmic function. We will be making use of this composite function in the derivation of fuzzy classifier. To this end, an algorithm is outlined here.

Algorithm for the Composite Fuzzy Classifier

1) Find the error vector between the training feature vector and test feature vector for the I^{th} user as:

$$e_{mj}^{l} = \left| x_{mj}^{l} - t_j \right|; \ m \neq h, \ j = 1, 2, \cdots, n$$

2) Compute the membership function vectors $\left(\mu_{m1}^{l}, \mu_{m2}^{l}, \cdots, \mu_{mn}^{l}\right)\left(\forall m = 1, 2, 3, \cdots, s\right)$ for the I^{th} user as follows:

$$\mu_{mj}^{l} = \mathrm{e}^{-\left(\left| x_{mj}^{l} - t_j \right|\right)}; \ m \neq h, \ j = 1, 2, \cdots, n$$

3) Compute the normed error vector $E_{mh}^{l}\left(\forall m, h = 1, 2, \cdots, s, \ m \neq h, \ j = 1, 2, \cdots, n\right)$ for the I^{th} user from:

$$E_{mh}^{l}(j) = t_F\left(e_{mj}^{l}, e_{hj}^{l}\right); \ m \neq h, \ j = 1, 2, \cdots, n$$

4) Compute the t-norm of a pair of membership functions, $M_{mh}^{l}\left(\forall m, h = 1, 2, \cdots, s, \ m \neq h, \ j = 1, 2, \cdots, n\right)$ for the I^{th} user as follows:

$$M_{mh}^{l}(j) = t_F\left(\mu_{mj}^{l}, \mu_{hj}^{l}\right), \ m \neq h, \ j = 1, 2, \cdots, n$$

5) Compute H_{mh}^{l} using Composite entropy function

$$H_{mh}^{l} = \log\left(\sum_{j=1}^{n} \left(E_{mh}^{l}(j) M_{mh}^{l}(j) \right)^{\alpha} \cdot \mathrm{e}^{\left(E_{mh}^{l}(j) M_{mh}^{l}(j) \right)^{\gamma}} \right)$$

6) Repeat Steps 1-4 for all users $l = (1, 2, \cdots, C)$ and if $k = \min \arg_l \{H_l\}$, then the test sample belongs to k^{th} user.

4. Methodology

The above Renyi entropy features are applied on the publicly available dataset from CMU.

For the evaluation of the keystroke dynamics based authentication system, we have used the following publicly available dataset:

CMU Keystroke Dynamics Benchmark Dataset [3]

Data is collected from 51 users in 8 sessions and 50 repetitions of the same password are recorded in each session. So, for each user there are 400 samples.

CMU benchmark dataset has keystroke features DD (Down-Down) time, UD (Up-Down) time and H (Hold) time. Each user has typed a 10-character password (".tie5Roanl"). For the evaluation of Renyi Entropy based features, we have used Hold and Up-Down times. Therefore, we have 21 features which include: 11 Hold Time values for 10 characters and an enter key, 10 Up-Down Time values for latencies between 11 key release and subsequent key press.

Half of the samples for each user (*i.e.* 200 samples) is used as the training data and the remaining half is used for positive testing. Each user is considered as both genuine and imposter user; thus facilitating 51 × 50 experiments.

For the classification, three classifiers are employed. The first one is Random Forest Classifier in which ensemble of decision trees is generated based on the training data. The second is two-class SVM classifier with a linear kernel. The third is the proposed Composite Fuzzy Classifier inspired from the Hanman Classfier [17].

To evaluate the performance of the derived features, error rates, viz., FAR (False Acceptance Rate), FRR (False Rejection Rate), EER (Equal Error Rate) and authentication accuracy are calculated for each of 51 × 50 experiments and reported.

5. Results of Implementation

Table 1 shows Error rates for different features derived above in terms of FAR, FRR, EER and Accuracy on CMU dataset with Random Forest as classifier. The best EER of 0.0152 is obtained with Sigmoid Renyi Function and the best accuracy of 0.9825 is obtained with Energy Renyi Feature.

Some of the features of Table 1 extracted from CMU database are classified using SVM and the results are given in Table 2. Here we get the best EER of 0.0279 with an accuracy of 0.9708 for Sigmoid Renyi Function.

The information set features derived from Renyi Entropy are applied on the Composite Fuzzy Classifier and the results are shown in Table 3. Here we get

Table 1. Comparison of results for different features with Random Forest classifier.

Feature	FAR	FRR	EER	Accuracy
Adaptive Renyi Function	0.0114	0.0254	0.0153	0.9824
Complement Renyi Function	0.0117	0.0258	0.0155	0.9820
Sigmoid Renyi Function	0.0112	0.0258	**0.0152**	0.9823
Complement Sigmoid Renyi Function	0.0118	0.0255	0.0155	0.9821
Energy Renyi Feature	0.0117	0.0247	0.0153	0.9825
Complement Energy Renyi Feature	0.0112	0.0271	0.0153	0.9818
Renyi Transform	0.0112	0.0261	0.0153	0.9822
Complement Renyi Transform	0.0116	0.0257	0.0153	0.9821
Modified Sigmoid Renyi Feature	0.0114	0.0256	0.0153	0.9823
Modified Complement Sigmoid Renyi Feature	0.0117	0.0277	0.0163	0.9812

Table 2. Comparison of results for different features with SVM.

Feature	FAR	FRR	EER	Accuracy
Adaptive Renyi Function	0.0187	0.0428	0.0283	0.9706
Complement Renyi Function	0.0191	0.0441	0.0290	0.9698
Sigmoid Renyi Function	0.0197	0.0412	**0.0279**	0.9708
Complement Sigmoid Renyi Function	0.0202	0.0436	0.0293	0.9694
Energy Renyi Feature	0.0218	0.0460	0.0312	0.9675
Complement Energy Renyi Feature	0.0201	0.0438	0.0291	0.9694
Renyi Transform	0.0185	0.0424	0.0285	0.9709
Complement Renyi Transform	0.0194	0.0430	0.0291	0.9701
Modified Sigmoid Renyi Feature	0.0183	0.0418	0.0272	0.9712
Modified Complement Sigmoid Renyi Feature	0.0218	0.0458	0.0310	0.9675

Table 3. Comparison of results for different features with Composite Fuzzy Classifier.

Feature	FAR	FRR	EER	Accuracy
Adaptive Renyi Function	0.0125	0.0190	**0.0144**	**0.9846**
Complement Renyi Function	0.0144	0.0186	0.0153	0.9837
Sigmoid Renyi Function	0.0113	0.0222	0.0148	0.9838
Complement Sigmoid Renyi Function	0.0166	0.0198	0.0167	0.9820
Energy Renyi Feature	0.0171	0.0268	0.0196	0.9786
Complement Energy Renyi Feature	0.0119	0.0241	0.0149	0.9827
Renyi Transform	0.0137	0.0180	0.0146	0.9844
Complement Renyi Transform	0.0141	0.0189	0.0152	0.9838
Modified Sigmoid Renyi Feature	0.0106	0.0248	0.0149	0.9831
Modified Complement Sigmoid Renyi Feature	0.0181	0.0241	0.0199	0.9793

the best performance with Adaptive Renyi Function for EER of 0.0144 and an accuracy of 0.9846.

Now we will compare the performance of Composite Fuzzy Classifier with SVM and Random Forest in terms of ROC curves. EER is computed by taking the mean of EERs from 51 × 50 experiments and their ROC curves. So, the comparison of ROC curves is shown for one experiment for the user 20 and imposter 11 of CMU dataset.

ROC curves for the above derived information set features for user number 20 with imposter 11 are shown in Figures 1-10.

In almost all the cases presented above, the proposed composite fuzzy classifier clearly outperforms SVM and Random Forest Classifiers in terms of both error rates and ROC curves.

6. Conclusions

We have presented an approach for the authentication of users based on keystroke dynamics using the Information set features derived from the adaptive

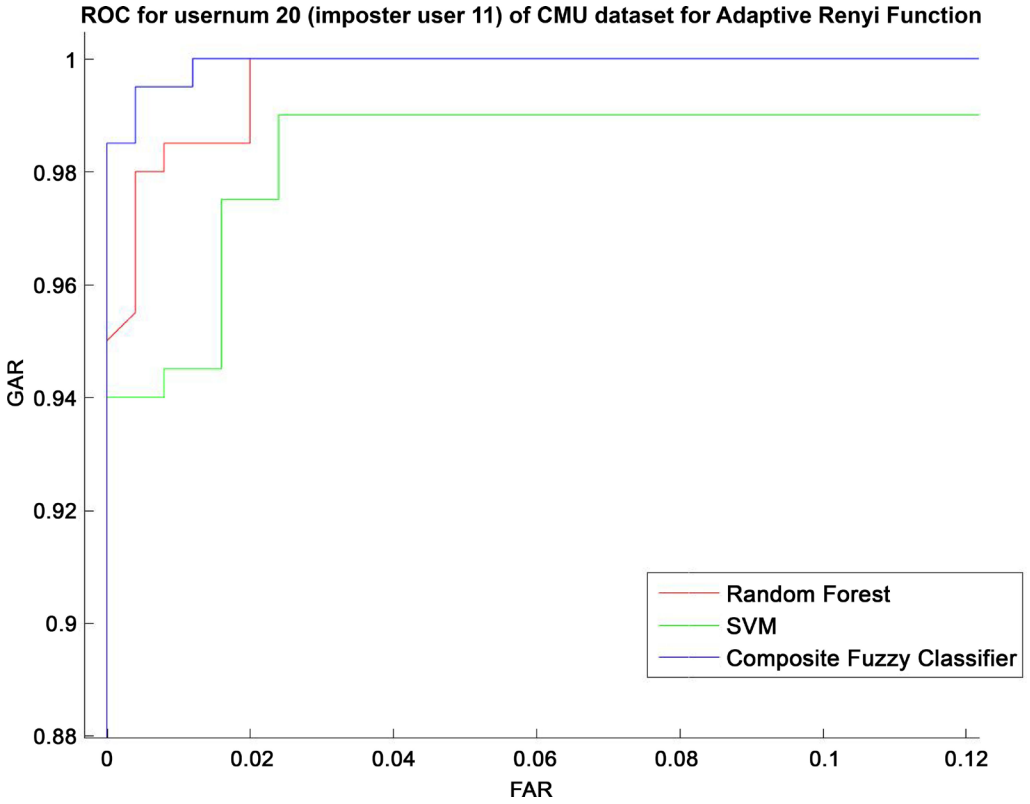

Figure 1. ROC for user 20 (imposter user 11) of CMU dataset for Adaptive Renyi Function.

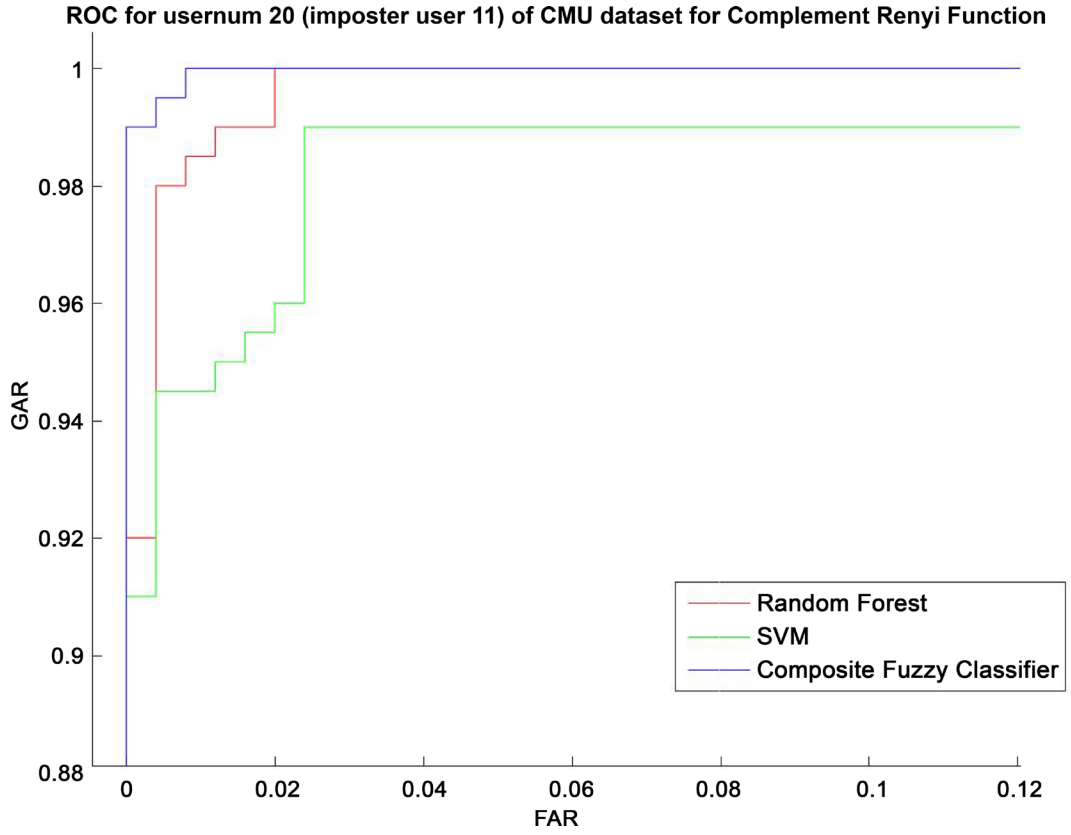

Figure 2. ROC for user 20 (with imposter user 11) of CMU dataset for Complement Renyi Function.

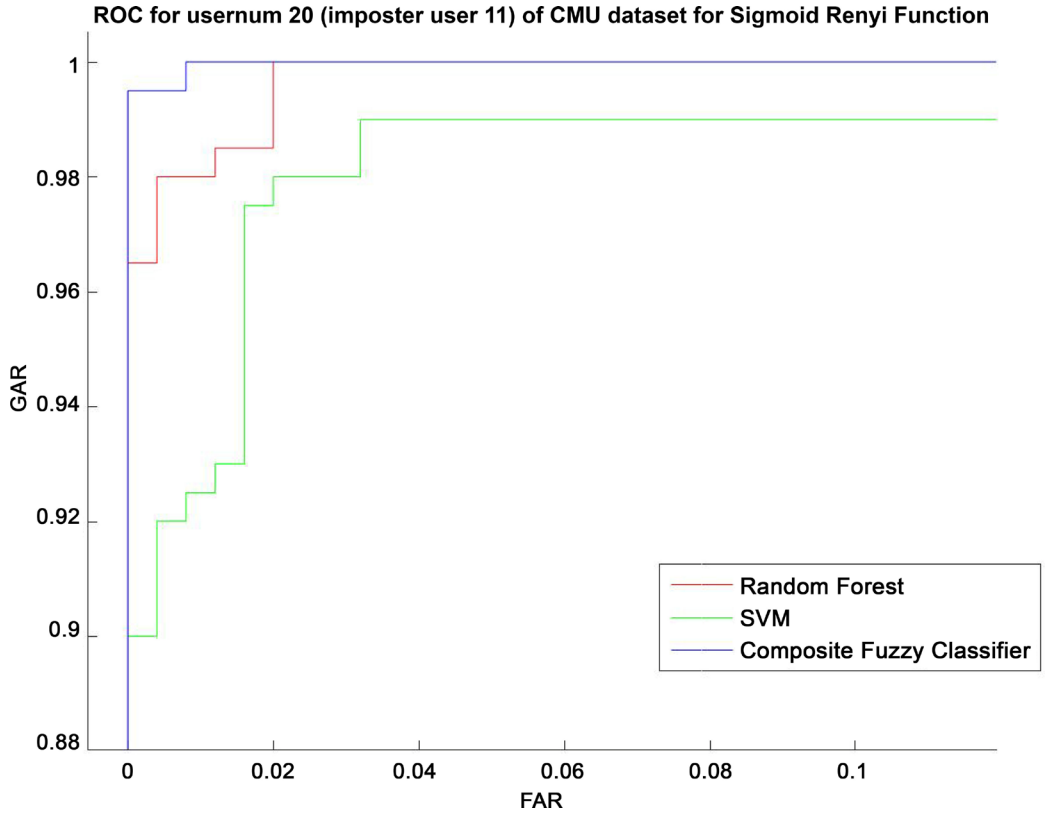

Figure 3. ROC for user 20 (with imposter user 11) of CMU dataset for Sigmoid Renyi Function.

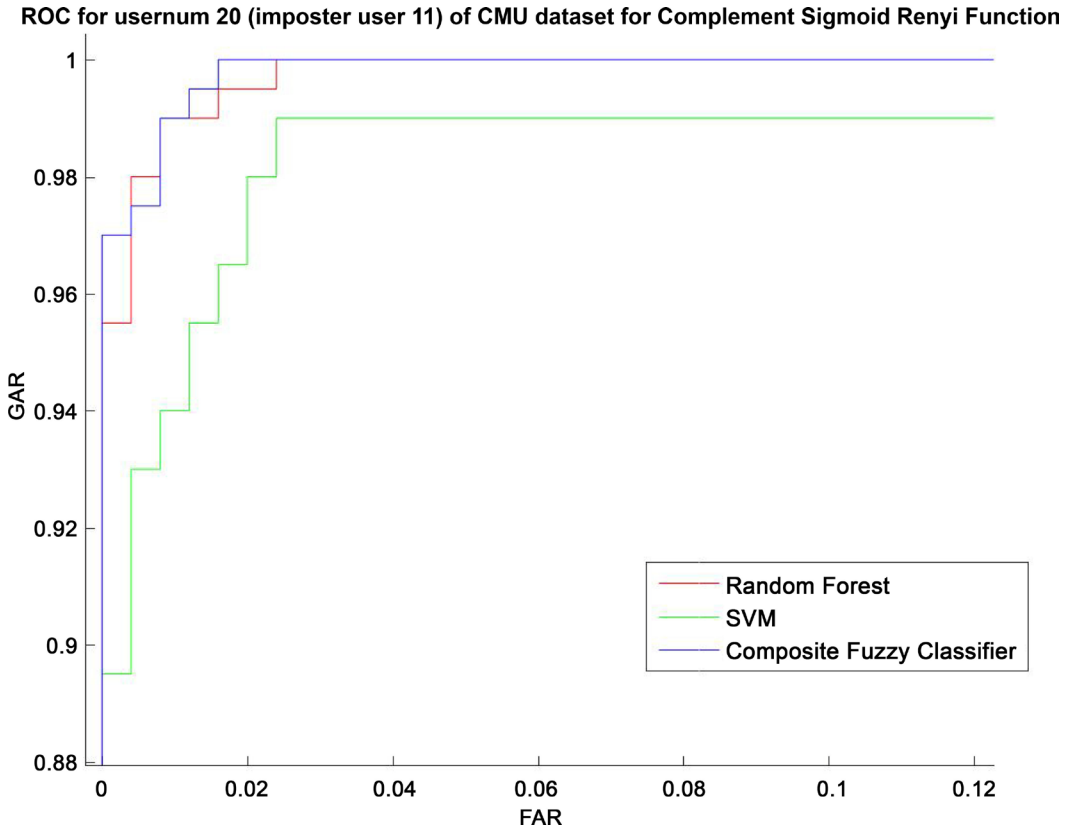

Figure 4. ROC for user 20 (imposter user 11) of CMU dataset for Complement Sigmoid Renyi Function.

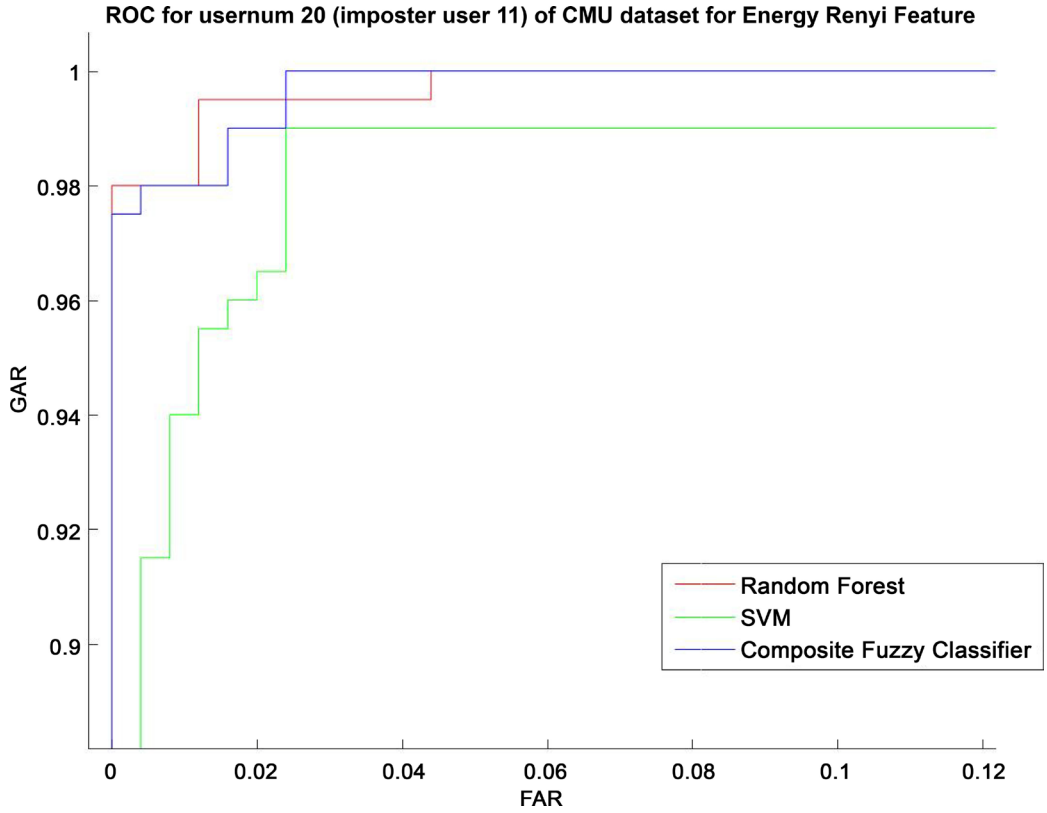

Figure 5. ROC for user 20 (with imposter user 11) of CMU dataset for Energy Renyi Feature.

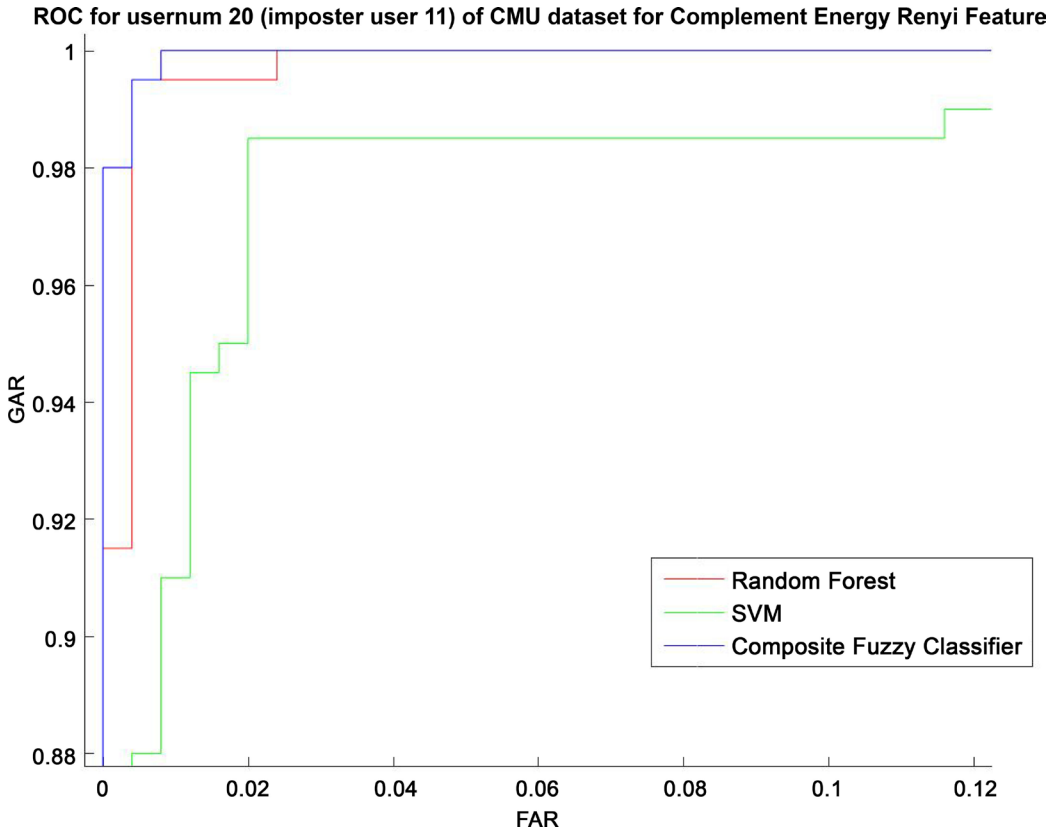

Figure 6. ROC for user 20 (with imposter user 11) of CMU dataset for Complement Energy Feature.

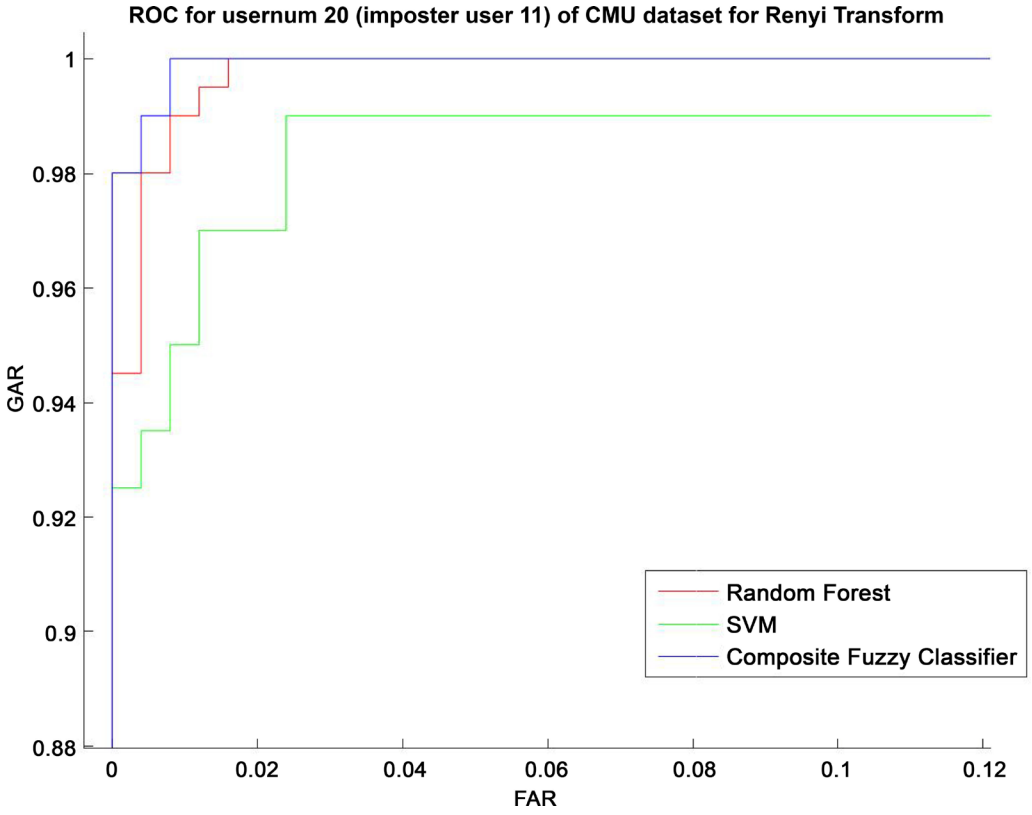

Figure 7. ROC for user 20 (with imposter user 11) of CMU dataset for Renyi Transform.

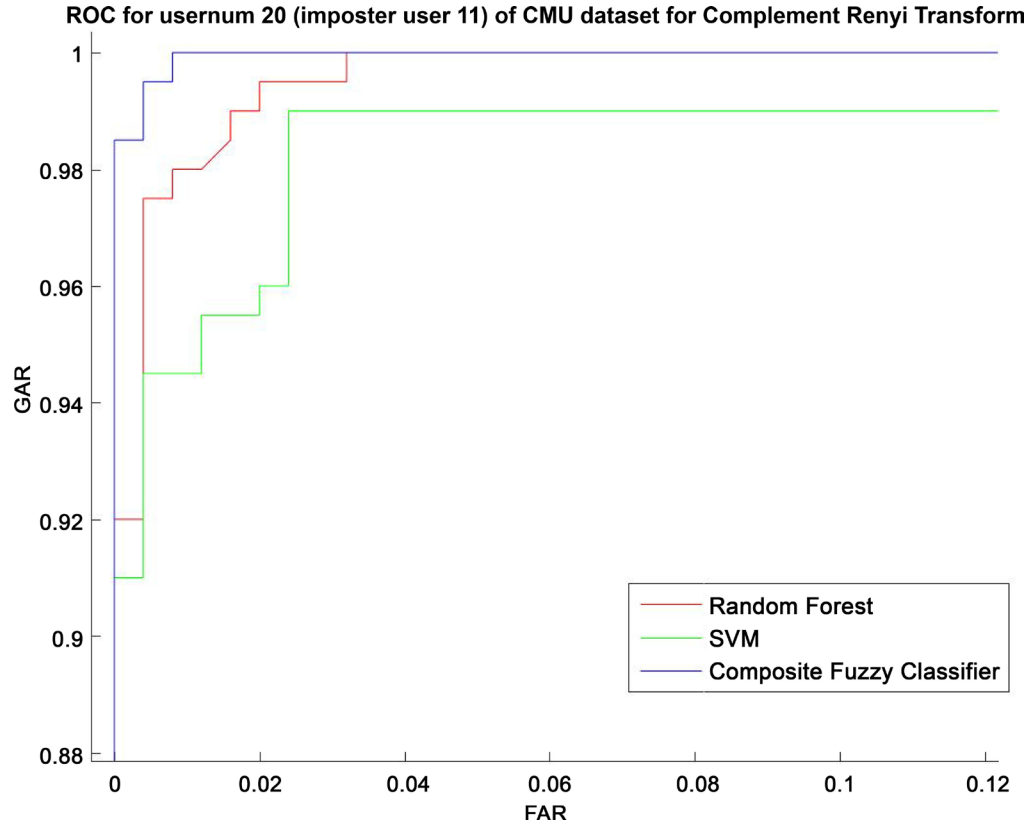

Figure 8. ROC for user 20 (imposter user 11) of CMU dataset for Complement Renyi Transform.

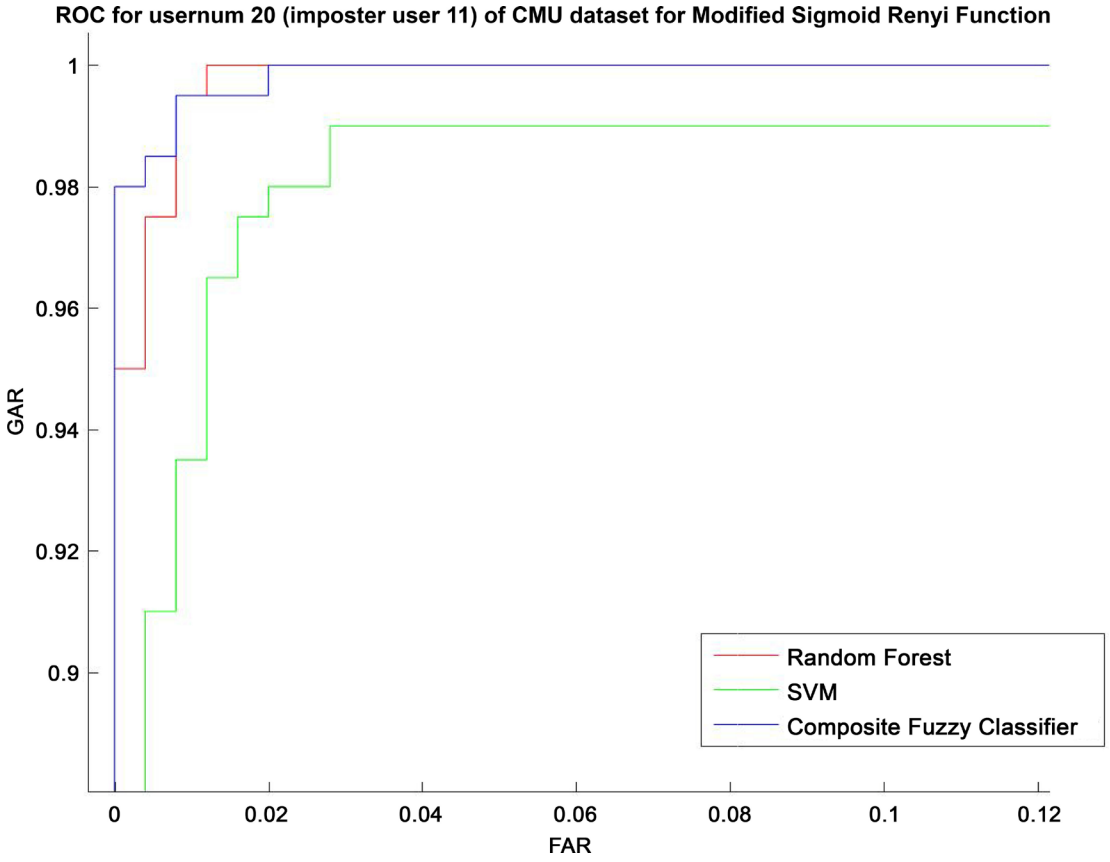

Figure 9. ROC for user 20 (with imposter user 11) of CMU dataset for Modified Sigmoid Renyi Function.

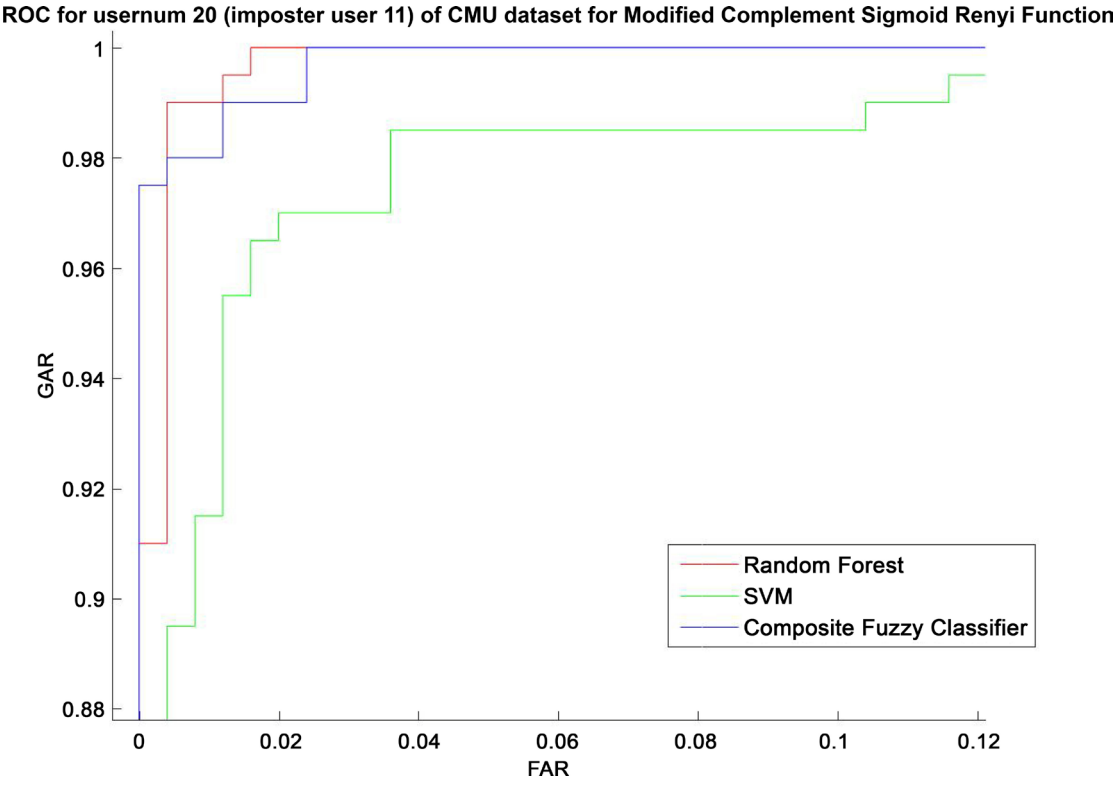

Figure 10. ROC for user 20 (with imposter user 11) of CMU dataset for Modified Complement Sigmoid Renyi Function.

Renyi entropy function by establishing its connection with Hanman-Anirban function. This in turn has paved the way in deriving several features in similar lines with the already existing information set features based on Hanman-Anirban entropy function. The feature vectors of a particular feature type corresponding to samples of each user are arranged in matrix form. Using columns as representing the spatial information component and rows as representing the temporal information, Two-Component information set (TCIS) features are derived. Thus TCIS features for all feature types are obtained.

For the development of composite entropy function the log function is applied on the Mamta-Hanman entropy function in which the product of the T-normed error value and T-normed-membership function value is considered as the information source value. Thus we have made use of the higher form of Mamta-Hanman entropy function. This composite entropy function is converted into a composite fuzzy classifier. Its performance is compared with that of Random forest classifier (Treebagger) and SVM. The best results are obtained with Adaptive Renyi entropy features using Composite fuzzy classifier. The results due to Random Forest and SVM are slightly inferior.

We hope the new features will find applications in different domains.

References

[1] Loy, C.C., Lim, C.P. and Lai, W.K. (2005) Pressure-Based Typing Biometrics User Authentication Using the Fuzzy ARTMAP Neural Network. *Proceedings of the Twelfth International Conference on Neural Information Processing* (*ICONIP* 2005), Taiwan, 30 October-2 November 2005, 647-652.

[2] Loy, C.C., Lai, W.K. and Lim, C.P. (2007) Keystroke Patterns Classification Using the ARTMAP-FD Neural Network. *Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kaohsiung, 26-28 November 2007, Vol. 1, 61-64. https://doi.org/10.1109/IIH-MSP.2007.218

[3] Killourhy, K.S. and Maxion, R.A. (2009) Comparing Anomaly-Detection Algorithms for Keystroke Dynamics. *IEEE/IFIP International Conference on Dependable Systems & Networks*, Lisbon, 29 June-2 July 2009, 125-134. https://doi.org/10.1109/DSN.2009.5270346

[4] Giot, R., El-Abed, M. and Rosenberger, C. (2009) GREYC Keystroke: A Benchmark for Keystroke Dynamics Biometric Systems. *IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, Washington DC, 28-30 September 2009, 1-6. https://doi.org/10.1109/BTAS.2009.5339051

[5] Montalvão Filho, J.R. and Freire, E.O. (2006) *Pattern Recognition Letters*, **27**, 1440-1446. https://doi.org/10.1016/j.patrec.2006.01.010

[6] Vural, E., Huang, J., Hou, D. and Schuckers, S. (2014) Shared Research Dataset to Support Development of Keystroke Authentication, *IEEE International Joint Conference on Biometrics* (*IJCB*), Clearwater, 29 September-2 October 2014, 1-8. https://doi.org/10.1109/BTAS.2014.6996259

[7] Wangsuk, K. and Anusas-Amornkul, T. (2013) *Procedia Computer Science*, **24**, 175-183. https://doi.org/10.1016/j.procs.2013.10.041

[8] Hosseinzadeh, D. and Krishnan, S. (2008) *IEEE Transactions on Systems, Man, and Cybernetics, Part C* (*Applications and Reviews*), **38**, 816-826.

[9] Eker, H. and Upadhyaya, S. (2015) Enhanced Recognition of Keystroke Dynamics Using Gaussian Mixture Models. *IEEE Military Communications Conference*, Tampa, 26-28 October 2015, 1305-1310.

[10] Deng, Y. and Zhong, Y. (2013) *ISRN Signal Processing*, **2013**, Article ID: 565183.

[11] The, P.S., Teoh, A.B.J., Ong, T.S. and Neo, H.F. (2007) Statistical Fusion Approach on Keystroke Dynamics. *Third International IEEE Conference on Signal-Image Technologies and Internet-Based System*, Shanghai, 16-18 December 2007, 918-923. https://doi.org/10.1109/SITIS.2007.46

[12] Thanganayagam, R. and Thangadurai, A. (2016) Hybrid Model with Fusion Approach to Enhance the Efficiency of Keystroke Dynamics Authentication. *Proceedings of* 3*rd International Conference on Advanced Computing, Networking and Informatics*, 85-96. https://doi.org/10.1007/978-81-322-2538-6_10

[13] Pisani, P.H., Giot, R., De Carvalho, A.C. and Lorena, A.C. (2016) *Computers & Security*, **60**, 134-153. https://doi.org/10.1016/j.cose.2016.04.004

[14] Ivannikova, E., David, G. and Hämäläinen, T. (2017) Anomaly Detection Approach to Keystroke Dynamics Based User Authentication. *IEEE Symposium on Computers and Communications* (*ISCC*), Heraklion, 3-6 July 2017, 885-889. https://doi.org/10.1109/ISCC.2017.8024638

[15] Mhenni, A., Rosenberger, C., Cherrier, E. and Ben Amara, N.E. (2016) Keystroke Template Update with Adapted Thresholds. 2*nd International Conference on Advanced Technologies for Signal and Image Processing* (*ATSIP*), Monastir, 21-23 March 2016, 483-488. https://doi.org/10.1109/ATSIP.2016.7523122

[16] Bhatia, A. and Hanmandlu, M. (2017) *Journal of Modern Physics*, **8**, 1557-1583. https://doi.org/10.4236/jmp.2017.89094

[17] Hanmandlu, M., *et al.* (2014) *Engineering Applications of Artificial Intelligence*, **36**, 269-286. https://doi.org/10.1016/j.engappai.2014.06.028

[18] Pal, N.R. and Pal, S.K. (1991) *IEEE Transactions on Systems, Man, and Cybernetics*, **21**, 1260-1270. https://doi.org/10.1109/21.120079

[19] Rényi, A., *et al.* (1961) On Measures of Entropy and Information. *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume* 1: *Contributions to the Theory of Statistics*, University of California Press, Berkeley, California, 20 June-30 July 1960, 547-561.

[20] Hanmandlu, M. and Das, A. (2011) *Defence Science Journal*, **61**, 415-430. https://doi.org/10.14429/dsj.61.1177

[21] Arora, P., Hanmandlu, M. and Srivastava, S. (2015) *Pattern Recognition Letters*, **68**, 336-342. https://doi.org/10.1016/j.patrec.2015.05.016

[22] Sayeed, F. and Hanmandlu, M. (2017) *Knowledge and Information Systems*, **52**, 485-507. https://doi.org/10.1007/s10115-016-1017-x

[23] Mamta and Hanmandlu, M. (2013) *Expert Systems with Applications*, **40**, 6478-6490. https://doi.org/10.1016/j.eswa.2013.05.020