Scientific Research Publishing

Journal of Modern Physics, 2017, 8, 1557-1583
http://www.scirp.org/journal/jmp
ISSN Online: 2153-120X
ISSN Print: 2153-1196

Keystroke Dynamics Based Authentication Using Information Sets

Aparna Bhatia*, Madasu Hanmandlu

Department of Electrical Engineering, Indian Institute of Technology, Delhi, India
Email: *aparna.bhatia@gmail.com, mhmandlu@gmail.com

How to cite this paper: Bhatia, A. and Hanmandlu, M. (2017) Keystroke Dynamics Based Authentication Using Information Sets. *Journal of Modern Physics*, **8**, 1557-1583.
https://doi.org/10.4236/jmp.2017.89094

Received: July 9, 2017
Accepted: August 20, 2017
Published: August 23, 2017

Abstract

This paper presents keystroke dynamics based authentication system using the information set concept. Two types of membership functions (MFs) are computed: one based on the timing features of all the samples and another based on the timing features of a single sample. These MFs lead to two types of information components (spatial and temporal) which are concatenated and modified to produce different feature types. Two Component Information Set (TCIS) is proposed for keystroke dynamics based user authentication. The keystroke features are converted into TCIS features which are then classified by SVM, Random Forest and proposed Convex Entropy Based Hanman Classifier. The TCIS features are capable of representing the spatial and temporal uncertainties. The performance of the proposed features is tested on CMU benchmark dataset in terms of error rates (FAR, FRR, EER) and accuracy of the features. In addition, the proposed features are also tested on Android Touch screen based Mobile Keystroke Dataset. The TCIS features improve the performance and give lower error rates and better accuracy than that of the existing features in literature.

Keywords

Information Set Theory, Two Component Information Set Features, Support Vector Machines (SVM), Random Forest, Convex Hanman-Anirban Entropy Function, Hanman Classifier, Convex Entropy Based Classifier

1. Introduction

Security is a concern since the advent of the computers. The need of robust and ubiquitous security systems is more apparent due to widespread use of Internet and rapidly growing online business transactions, e-banking, shopping, social interactions, emails to name a few. User authentication involving both identifi-

cation and verification has become a necessity before the access to system resources is allowed. The most common user authentication system till date employs username and password/PIN. Irrespective of whether a user chooses a very easy password or forgets any password he has chosen; the system may be prone to misuse in either case. It is possible to steal or hack the most difficult password by means of brute force methods. Use of biometrics for personal authentication is becoming more acceptable these days because it is convenient to use and there is no issue of getting lost like smart cards and no problem of getting forgotten like passwords/PINs. Biometrics deals with physiological or behavioral human traits for authentication of a user. Biometrics provides significant security compared to username/password, smartcards etc. Among biometric traits, keystroke dynamics is most convenient since keyboard is available in most of the computer systems and does not require a special device like other biometric modalities such as fingerprint, palmprint etc. Keystroke dynamics based authentication is concerned with analyzing the human typing rhythm and behavior. Further keystroke dynamics is difficult to conceal and disguise just as human behavior is difficult to copy. Keystroke Dynamics can also be implemented on a network or distributed architecture.

1.1. Background Research

Keystroke dynamics based authentication system is dependent on the individual typing pattern. It is mainly based on how a user types rather than what the user types on keyboard. It measures human typing characteristics which are shown to be unique to an individual and difficult to be copied. In keystroke dynamics, there are mainly two metrics, Dwell Time which is how long a key is pressed and the other is Flight Time which is how long it takes the user to move from one key to the other. As the user types, an application running on the system captures the keystroke dynamics features flight time and dwell time.

There are some publicly available keystroke datasets. Most of these datasets prefer static text entry, for which a user is asked to type a predetermined text string. Some of the static entry dataset are from: Killorhy and Maxion [1], Giot *et al.* [2], Loy *et al.* [3] [4]. Very few datasets like Biochaves [5], Clarksons University Keystroke Dataset [6] are based on free text.

Keystroke Dynamics features mainly include: Keystroke Latencies, Dwell Time and Flight Time. Gaines *et al.* [7] employ inter-key latencies from 87 lowercase letters to compute the means of the keystroke latencies and check their similarity. Young and Hammon [8] have used keystroke latency, keystroke pressure and total time to type as features for their experiments. They have built a template using these features. To authenticate a test feature vector, the mean timing vector and inverted covariance of timing vectors are computed [9], and then compared statistically with the test timing vector. Joyce & Gupta [10] have augmented the login process by asking for user's first name and last name in addition to the usual procedure of asking login name, password and latency infor-

mation as feature subset. For recording the timestamp, special scan codes are used in the interrupt handler of the standard keyboard. When two keys are pressed such that the first key is not yet released and the second gets pressed, then a negative time measurement occurs which is a limitation. To overcome this, a modified latency measurement is suggested in [11]. A combination of key hold time and digraph latency metrics is used in [12] to reduce error drastically. The features used in [6] are of four types: key code (ASCII code of the key being pressed) and three timing features that include: Down-Down Time (DD), Up-Down Time (UD) and Down-Up Time (DU). The first two timing features are used to denote the inter-key latencies and third feature indicates the hold-time.

For authentication that involves both identification and verification of a user by keystroke dynamics based system, many classifiers have been used. They are divided into three broad categories, viz., statistical methods, neural networks and pattern recognition based techniques.

The statistical methods related to the first category employ statistical tools on basic keystroke features and apply distance metric to authenticate a user. The initial work on Keystroke Dynamics by Gaines *et al.* [7] involves t-test on diagraph features to check the similarity of mean vectors and covariance matrices on two multivariate normal populations giving FAR of 0% and FRR of 4%. But this is impossible to achieve in real life situation where the number of users is very less. Umphress and William [13] identify a user by comparing the keystroke latencies and digraphs of the test sample with the reference profile data comprising the mean keystroke latency and average time to press in the two consecutive keys. A confidence score is specified to achieve FAR of 17% and FRR of 30%. Joyce and Gupta [10] have developed a mean reference signature consisting of a set of four vectors of keystroke latencies for username, password, first name and last name. The norm is computed between the test keystroke pattern and the reference signature and then the user authentication is done based on some predefined threshold. By this FAR of 0.25% and FRR of 16.67% are achieved. Teh *et al.* [14] have proposed a statistical fusion approach for keystroke dynamics based recognition system and they authenticate a user using the weighted sum of Gaussian scores and Direction Similarity Measure based scores.

We now detail out the neural network based approaches under the second category. Giroux *et al.* [15] have used keypress ratios as a measure of authentication and a dedicated Artificial Neural Network (ANN) is employed for the authentication of a user. A function f: $R^{m-1} \rightarrow \{-1, 1\}$ is learned from ANN, where $x \in R^{m-1}$ denotes the $m - 1$ keypress interval timing ratios for m-character password and $f(x) = [-1, 1]$ indicates whether the input keypress interval ratios correspond to that user or not. For every individual, a feed-forward ANN is trained with back-propagation, resulting in weights that are subsequently used for authentication. Bleha *et al.* [16] have used linear perceptron to authenticate the users and reported error rates, FAR and FRR of 9% and 8% respectively. The use of

two separate orthogonal digraph components – Keystroke duration and Keystroke latency are found to provide significant predictive power with Back-Propagation Model [12]. To attain IPR of 0% and FAR of 11.5%, a preprocessing step is performed.

The pattern recognition based techniques falling under the third category are now discussed. Support Vector machine (SVM) based on keystroke latency in [17] gives FAR of 0.02 and FRR of 0.1 for 10 users. The keystroke latency and key hold time are used as features for k-nearest neighbor classifier in [18]. The classical pattern recognition based algorithms such as back propagation with sigmoid transfer function, sum of products, hybrid sum of products, Bayes' decision theory and Potential Function are used in [19] for combining key hold time and interkey latencies. Among various pattern recognition techniques used in [19], potential function gives the best results with FAR and FRR of 0.7% and 1.9% respectively.

1.2. Motivation for the Present Work

From the literature survey, it can be seen that most of the approaches on keystroke dynamics are carried out on the created datasets and they report results either on desktop or mobile but not both. It is difficult to compare the performance of different approaches due to lack of common benchmark dataset. So, we have tested the proposed approach on the benchmark datasets under both desktop and mobile environments and the results obtained are found to be superior to the best so far.

The organization of the paper is as follows: Section 2 presents the information set (IS) and some of its properties. It also formulates the IS based features and higher form of IS features. Section 3 develops an algorithm for the two-way information set approach. Section 4 describes the databases for the present work and Section 5 discusses the results of implementation. Section 6 gives the conclusions and the future work.

2. An Introduction to Information Set

A fuzzy set deals with vagueness or fuzziness [20]. It is characterized by a membership function (MF) that maps the information source values to the degree of association in the range (0, 1). The MF of x_i in a fuzzy set (F) is denoted by $\mu_F(x_i)$. Given a collection of attribute values $X = \{x_1, x_2, \cdots, x_n\}$, F is a set of ordered pairs $\{(x_1, \mu_F(x_1)), (x_2, \mu_F(x_2)), \cdots, (x_n, \mu_F(x_n))\}$. A fuzzy set suffers from some drawbacks [21]: i) The values of MF are separate from the information source values. There is no way to link the two into a single entity. ii) MF doesn't provide the overall fuzziness/vagueness of F but only the degree of association of every information source value to a vague concept, and iii) The time varying information source values are not easily represented in MF. To eliminate these drawbacks of a fuzzy set, Hanmandlu and his co-works have developed Information set theory which can be found in [21]-[27] based on the information theo-

retic entropy function christened as Hanman-Anirban entropy function. The properties of information sets given later in this section will highlight the power of information sets.

Our primary goal being the representation of overall uncertainty in keystroke dynamics, we are inclined to investigate the suitability of the information set based features. We will now discuss how a fuzzy set paves the way for the information set while representing the uncertainty in its elements using an entropy function.

2.1. Information Set Concept

Consider a set of keystroke timing features $T = \{T_{ij}\}$ where T_{ij} is the j[th] feature in i[th] keystroke sample. When a set of keystroke timing features is fitted with a membership function, denoted by $\{\mu_{ij}\}$, a pair of keystroke timing value and its membership function forms an element in a fuzzy set. Information set connects the two components of each pair into a single entity called the information value using the Hanman-Anirban Entropy function [22] which has the facility to represent both probabilistic and possibilistic uncertainties. The probabilistic uncertainty in a fuzzy set is defined by Hanman-Anirban entropy function having a polynomial in its exponential gain function as:

$$H = \sum_i \sum_j p_{ij} \mathrm{e}^{-\left(ap_{ij}^3 + bp_{ij}^2 + cp_{ij} + d\right)} \tag{1}$$

where $H_{ij} = p_{ij} \mathrm{e}^{-\left(ap_{ij}^3 + bp_{ij}^2 + cp_{ij} + d\right)}$,
and a, b, c and d are real valued parameters which need to be selected appropriately. It may be noted that p represents the probabilities. As shown in [23] that the possibilistic uncertainty is a better representation of uncertainty than the probabilistic uncertainty given by Equation (1). Moreover, the number of probabilities is limited in the context of keystroke dynamics; this is the reason we are bent upon exploring the possibilistic uncertainty.

To bring Equation (1) into the information set domain, let us call the keystroke timing features T_{ij} as the information source values. We then replace the probability p with T_{ij} in Equation (1) and convert the exponential gain function into the Gaussian membership function by selecting the parameters as $p_{ij} = T_{ij}$, $a = 0$, $b = \frac{1}{2\sigma^2}$, $c = -\frac{2T_{ref}}{2\sigma^2}$, $d = \frac{T_{ref}^2}{2\sigma^2}$ leading to:

$$H_{ij} = T_{ij} \mathrm{e}^{-\left\{\frac{\left(T_{ij} - T_{ref}\right)^2}{2\sigma^2}\right\}} = T_{ij}\mu_{ij} \tag{2}$$

A more general entropy function is presented by Mamta and Hanmandlu in [24]. This entropy function not only converts the exponential gain into the generalized Gaussian membership function with an exponent power of β but also modifies the information source values with a power of α. This is defined as:

$$H = \sum_i \sum_j p_{ij}^{\alpha} \mathrm{e}^{-\left(cp_{ij}^{\gamma} + d\right)^{\beta}} \tag{3}$$

where $H_{ij} = p_{ij}^{\alpha} \mathrm{e}^{-\left(cp_{ij}^{\gamma}+d\right)^{\beta}}$

By taking $\gamma = 1$ and $c = \frac{1}{2\sigma^2}$, $d = -\frac{T_{ref}}{2\sigma^2}$, Equation (3) becomes

$$H_{ij} = T_{ij}^{\alpha} \mathrm{e}^{-\left\{\frac{\left(T_{ij}-T_{ref}\right)}{2\sigma^2}\right\}^{\beta}} = T_{ij}^{\alpha} \mu_{ij}^{\beta} \tag{4}$$

The product of Information source value and membership function is termed as the information value and this is more general than the one in Equation (2). The sum of all information values, $\sum_i \sum_j H_{ij}$ gives the effective information. In this work, we are using only the information value as a feature.

Definition of Information Set: A set of information values $H = \left\{T_{ij}^{\alpha} \mu_{ij}^{\beta}\right\}$ is called the information set such that each information value is a product of the information source value and the corresponding membership function value. The values of α and β need to be selected appropriately.

2.2. Some Properties of Information Sets

The properties of information sets are presented in [25]. Following are the important properties of Information Sets:

1) The membership function can be empowered to act as an agent with the capabilities that are beyond the scope of a fuzzy set. For example, the complement of a membership function can be an agent. Any intuitionist membership function can also be a contender. The membership function can be formed from other information source values not associated with the same fuzzy set. Thus, an agent extends the scope of a fuzzy set.

2) The higher form of information sets called transforms can be derived based on the information values. This is shown in the sequel.

3) The information set arises out of representing the varying information source values in either time or space. For example, a variation in the keystroke data within a sample gives the spatial information values whereas the variation in keystroke timings over a number of samples gives the temporal information values.

4) Information set can represent both probabilistic and possibilistic uncertainties. To represent the probabilistic uncertainty, frequencies of occurrence of the information source values called the probabilities are considered but for the possibilistic uncertainty, attribute values like keystroke timing values are considered.

2.3. Derivation of Information Set Based features

We will now derive the information set based features. The use of basic information set features like sigmoid and energy appears in [26]. It is important to note that our unit of information is either the information value $T_{ij}^{\alpha} \mu_{ij}^{\beta}$ or the complement information value $T_{ij}^{\alpha} \overline{\mu}_{ij}^{\beta}$.

a) *Information Value*

The basic information values $H_{ij} = T_{ij}^{\alpha} \mu_{ij}^{\beta}$ can also be used as one of the fea-

tures in our study. The membership function is taken to be Gaussian with $\beta = 2$:

$$\mu_{ij}^{\beta} = \mathrm{e}^{-\frac{\left(T_{ij}-T_{ref}\right)^2}{2\sigma^2}} \tag{5}$$

where the reference, $T_{ref} = T_{avg}$ is the average of T_{ij} values. One can take any value such as mean, maximum and median for the reference.

b) *Complement Information Value*

As per the second property of information sets stated above the complement of membership function, *i.e.*, $\overline{\mu}_{ij}$ is found to be useful as an agent which is an empowered membership function with an extended scope as compared to that of a fuzzy set. As a result, the complement information value $T_{ij}^{\alpha}\overline{\mu}_{ij}^{\beta}$ serves as the feature, given by

$$H_{ij} = T_{ij}^{\alpha}\overline{\mu}_{ij}^{\beta} \tag{6}$$

where $\overline{\mu}_{ij} = 1 - \mu_{ij}$. Note that the complement membership function has its domain out of the fuzzy set.

c) *Energy features*

As the information value depends on the membership function empowered as an agent, we can generate different kinds of information values by changing the agent. To generate Energy feature, the agent is taken as μ^2:

$$E_{ij} = T_{ij}^{\alpha}\left\{\mu_{ij}^{\beta}\right\}^2 \tag{7}$$

So, the complement energy feature is:

$$\overline{E}_{ij} = T_{ij}^{\alpha}\left\{\overline{\mu}_{ij}^{\beta}\right\}^2 \tag{8}$$

d) *Sigmoid feature*

According to the first property of information set, information value ($T_{ij}^{\alpha}\mu_{ij}^{\beta}$) considering it as a unit of information can be modified by applying some function like sigmoid function. Note that the effectiveness of the information value (feature) gets enhanced with the application of this function. So, the modification of information value using the sigmoid function leads to the sigmoid feature defined as:

$$S_{ij} = \frac{1}{1+\mathrm{e}^{-T_{ij}^{\alpha}\mu_{ij}^{\beta}}} \tag{9}$$

e) *Multi Quadratic feature*

The multi-quadratic function either increases or decreases monotonically from the center. Using this function, the membership function is computed as:

$$\mu_{ij}^{MQ} = \sqrt{T_{ij}^2 + f_h^2} \tag{10}$$

where f_h^2 is a fuzzifier given by $f_h^2 = \dfrac{\sum_i \sum_j \left(T_{ij} - T_{avg}\right)^4}{\sum_i \sum_j \left(T_{ij} - T_{avg}\right)^2}$. The multi quadratic information value is computed as $T_{ij}^{\alpha}\mu_{ij}^{M}$.

f) Inverse Multi Quadratic feature

Inverse multi-quadratic function is the reverse of multi-quadratic function. Membership function for the inverse multi quadratic feature is given by

$$\mu_{ij}^{invMQ} = 1/\sqrt{T_{ij}^2 + f_h^2} \tag{11}$$

The inverse multi-quadratic information value is therefore $T_{ij}^{\alpha}\mu_{ij}^{invMQ}$.

2.4. Higher Form of Information sets

So far, we have utilized the basic information values for deriving different features. We will now derive higher form of information set based features. This requires us to consider the adaptive Mamta-Hanman entropy function in which the parameters of the exponential gain function are assumed to be variables. Some important properties of this adaptive entropy function are relegated to **Appendix A**.

a) Hanman Transform

Hanman Transform is a higher form of information derived from the adaptive Mamta-Hanman entropy function in [24]. The use of this transform appears in [27]. The idea of this transform is to use the first-level information values in getting the second-level information values. Thus, this transform is intended to get a better representation of the uncertainty in the information source values. The Hanman Transform (HT) is defined as

$$HT = \sum_i \sum_j T_{ij}^{\alpha} \mathrm{e}^{-\mu_{ij} T_{ij}} \tag{12}$$

where $HT_{ij} = T_{ij}^{\alpha} \mathrm{e}^{-\mu_{ij} T_{ij}}$

Proof: By taking $p_{ij} = T_{ij}$, $c_{ij} = \mu_{ij}$, $d_{ij} = 0$, $\gamma = 1$ and $\beta = 1$ in (3) we obtain (12).

Note that the exponential gain function has its argument as the first-level information value and after evaluation using Equation (12) we get the second-level information value. This is called transform because the original Information source value T_{ij} is modified by the information value, $T_{ij}\mu_{ij}$.

The Complement Hanman Transform is easily obtained by setting $c_{ij} = \bar{\mu}_{ij}$ in the above proof as

$$\overline{HT_{ij}} = T_{ij}^{\alpha} \mathrm{e}^{-\bar{\mu}_{ij} T_{ij}} \tag{13}$$

b) Shannon Transform

Shannon Transform is an offshoot of Hanman Transform as it can only be derived from the Hanman Transform and its features are shown to be useful in the face recognition in [22]. The Shannon transform (Sh) features are computed from:

$$Sh_{ij} = -T_{ij}^{\alpha} \log\left(\mu_{ij} T_{ij}\right) \tag{14}$$

Proof:

Again, we resort to the adaptive Mamta-Hanman entropy function (3) and set $c_{ij} = \mu_{ij}$, $d_{ij} = -1$ and $\beta = 1$ leading to

$$PP = \sum_i \sum_j T_{ij}^{\alpha} \mathrm{e}^{-\left(\mu_{ij} T_{ij} - 1\right)} \tag{15}$$

where $PP_{ij} = T_{ij}^{\alpha} \mathrm{e}^{-\left(\mu_{ij} T_{ij} - 1\right)}$

This is Pal-Pal transform. This can be shown to be equivalent to what we term as the non-linear Shannon transform in Equation (14) where the logarithmic function is operating on the information values. In some applications, the use of complement of μ_{ij} in the transform improves its effectiveness. The Shannon inverse transform where the evaluation of the information source values is based upon the complement agent is expressed as:

$$\overline{Sh_{ij}} = -T_{ij}^{\alpha} \log\left(\overline{\mu}_{ij} T_{ij}\right) \tag{16}$$

In the above transforms, Gaussian membership function as defined in Equation (5) is best suited. These transforms can have realistic applications in social networks though not attempted so far. For example, we gather information about an unknown person of some interest to us. This is the first-level of information and then evaluate him again to get the second-level of information camped with the first-level of information. They can be used to evaluate not only the information source values but also the membership function values to see whether the selected membership function is appropriate.

c) *Composite Transform*

For creating sigmoid and energy features, we have considered the basic information value $T_{ij}^{\alpha} \mu_{ij}^{\beta}$ as the unit of information. But to create the composite transform consider the Hanman transform feature $HT_{ij} = T_{ij}^{\alpha} \mathrm{e}^{\mu_{ij} T_{ij}}$ as the unit of information and apply the log function on it leading to the composite transform given by

$$CT_{ij} = \log\left(T_{ij}^{\alpha} \mathrm{e}^{\mu_{ij} T_{ij}}\right) \tag{17}$$

In fact, this is the *ij* component of the following transform:

$$CT = \log\left(\sum_i \sum_j T_{ij}^{\alpha} \mathrm{e}^{\mu_{ij} T_{ij}}\right) \tag{18}$$

By interchanging log and exponential function we can formulate yet another composite transform as given by

$$CT = \exp\left(-\sum_i \sum_j \mathrm{e}^{T_{ij}^{\alpha} \log\left(\mu_{ij} T_{ij}\right)}\right) \tag{19}$$

As can be noted that the difference between Equations (18) and (19) is that in the former case log function is applied on the Hanman transform whereas in the latter case the exponential function is applied on the Shannon transform. In this paper, we have shown the results of Equation (18).

The Complement Composite Transform is easily obtained by considering the complement Hanman Transform as the unit of information and applying the log function on it. It is given by

$$\overline{CT}_{ij} = \log\left(T_{ij}^{\alpha} \mathrm{e}^{\overline{\mu}_{ij} T_{ij}}\right) \tag{20}$$

d) *Convex Hanman-Anirban Entropy Function*

Let ϕ_i be convex and twice differentiable function and ψ is convex, twice differentiable and strictly increasing function. Then the generalized mean can be written as: [28]

$$G_m = \psi^{-1}\left(\sum_i^n w_i \phi_i\left(x_i\right)\right) \tag{21}$$

where w_i are such that $0 \le w_i \le 1$, $\sum w_i = 1$. Assuming

$$\phi_i\left(x\right) = x\mathrm{e}^x, \forall i = 1, 2, \cdots, k \tag{22}$$

Its double derivative is therefore

$$\phi_i''\left(x\right) = \left(x+2\right)\mathrm{e}^x > 0, \forall x \in R. \tag{23}$$

Supposing $\psi\left(x\right) = \mathrm{e}^x$ then

$$\psi^{-1}\left(x\right) = \log x, \forall x > 0 \in R. \tag{24}$$

If $w_i = \frac{1}{n}$ then $\sum w_i = 1$. Substituting (24) in (21) yields the convex entropy function

$$H_c = \log\left(\sum_i w_i x_i \mathrm{e}^{x_i}\right) = \frac{1}{n}\log\left(\sum_i x_i \mathrm{e}^{x_i}\right) \tag{25}$$

If we take $x_i = \mu_i T_i$ which is the unit of information in Equation (25), we obtain convex Hanman-Anirban entropy function:

$$H_c = \frac{1}{n}\log\left(\sum_i \mu_i T_i \mathrm{e}^{\mu_i T_i}\right) \tag{26}$$

We can find its use in the design of or to modify a classifier. Here we use it to modify the Hanman classifier.

3. Feature Extraction and Classifier Design

3.1. The Two-Component Information Set (TCIS)

We have at our disposal several samples of keystroke dynamics for each user. To calculate the membership function, we have adopted Two-Component information set approach. In this approach, the temporal information I_1 is the first component for which the membership function μ_1 is computed using all the training samples. The spatial information I_2 is the second component for which the membership function μ_2 is computed using all the features in a single sample. Concatenation of these two information components results in Two-Component Information Set features denoted by *I*. A flowchart shown in **Figure 1** explains how the features are computed in both training and test parts. The features from these parts go to a classifier for the authentication of keystroke sample.

Algorithm:

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

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

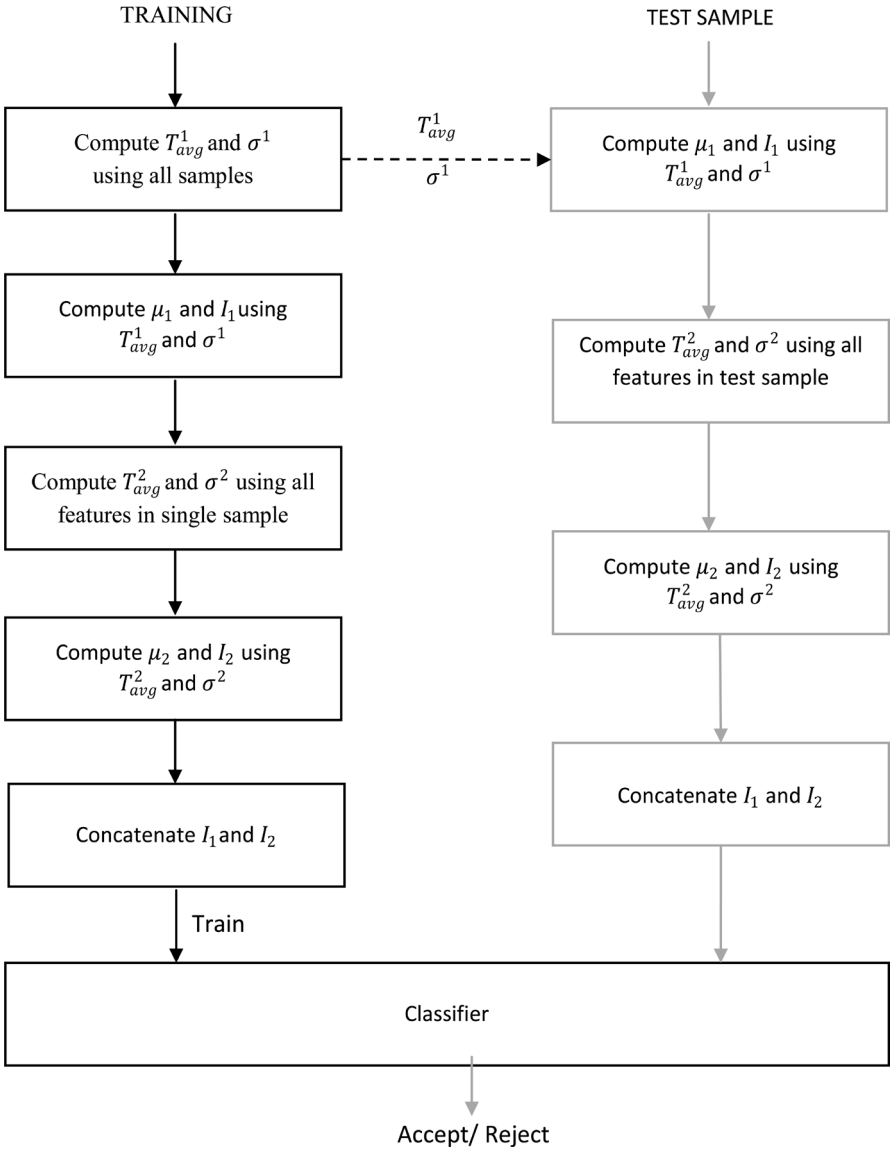

Figure 1. Flow Chart for authentication using TCIS features.

Step 3: For each training sample, compute μ^1 using T^1_{avg} and σ^1 and then compute μ^2 using T^2_{avg} and σ^2. These membership functions along with T give us two components, $I_1 = \left\{\mu^1_{ij} T_{ij}\right\}$ and $I_2 = \left\{\mu^2_{ij} T_{ij}\right\}$.

Step 4: Concatenate I_1 and I_2 and generate new features such as Information, Energy, Sigmoid, Hanman Transform etc. Then train SVM/Random Forest classifier or Convex Entropy Based Classifier using these features.

Step 5: For each test sample, compute I_1 using T^1_{avg} and σ^1 computed in Step 1.

Step 6: Compute mean $\left(T^2_{avg}\right)$ and variance $\left(\sigma^2\right)$ of all the features for the test sample. Compute I_2 using T^2_{avg} and σ^2.

Step 7: Concatenate I_1 and I_2 to obtain I and use the new features for the classification using SVM/ Random Forest classifier or Convex Entropy Based Classifier.

3.2. Design of Hanman Classifier (HC) Using Convex Entropy Function

As I is a feature vector, let us denote the training feature vector of rth sample of lth user by $P_l(r, k)$ and $Q(k)$ be the test feature vector where k refers to the kth feature value. The training and testing feature vectors are subjected to min-max normalization. In view of Equation (25), the test feature vector is rewritten as:

$$H_{ts} = \frac{1}{n}\log\left(\sum_k Q(k)\mathrm{e}^{Q(k)}\right) \tag{27}$$

Similarly, each training feature vector can also be denoted in the above form as:

$$H_t = \frac{1}{n}\log\left(\sum_k P(r,k)\mathrm{e}^{P(r,k)}\right) \tag{28}$$

a) ***Use of Conditional Entropy Function***

The conditional Hanman-Anirban entropy termed here as conditional possibility, c poss of a test feature vector $Q(k)$ given the training feature vector $P_l(r, k)$ is expressed by following [25] as:

$$cposs\left(P_l(r,k)/Q(k)\right) = \left\{P_l(r,k) - Q(k)\right\} = \left\{e_{ij}\right\} \tag{29}$$

The conditional possibility of intersection of two training feature vectors given the test feature vector can be written as:

$$\begin{aligned}&\left\{cposs\left(P_l(i,k)\cap P_l(j,k)\middle|Q(k)\right)\right\}\\&=\left(P_l(i,k)-Q(k)\right)\cap\left(P_l(j,k)-Q(k)\right)\\&=\left\{e_{il}(k)\cap e_{jl}(k)\right\}=\left\{t_F\left(e_{il}(k),e_{jl}(k)\right)\right\}=\left\{E_{ij,l}(k)\right\}\end{aligned} \tag{30}$$

As t-norm being the conjunction operator it gives the minimum difference between any two vectors in (30) where we have used Frank t-norm for t_F as it is found to be most effective [24]. It is given by

$$t_F = \log_q\left[1+\frac{\left(q^{e_{il}(k)}-1\right)\left(q^{e_{jl}(k)}-1\right)}{q-1}\right];\ k=1,2,\cdots,V \tag{31}$$

We call $E_{ij,l}(k)$ as the normed error vector as it is the result of applying t-norm on the pairs of two error vectors. We now invoke the convex entropy function for the representation of uncertainty in the normed error vectors.

$$h_{ij}(l) = \log\left\{\sum_{k=1}^{M} E_{ij,l}(k)\mathrm{e}^{E_{ij,l}(k)}\right\} \tag{32}$$

In order to improve the above convex entropy function, we convert it into parametric form:

$$h_{ij}(l) = \log\left\{\sum_{k=1}^{M} E_{ij,l}(k)^{\gamma}\,\mathrm{e}^{\rho E_{ij,l}(k)}\right\} \tag{33}$$

where γ and ρ are the parameters. The proof of (33) which is no longer a convex function can be given as follows: This entropy follows from Mamta-Hanman entropy function with proper substitution of parameters. Taking loga-

rithm of this entropy function $\sum_{k=1}^{M} E_{ij,l}(k)^{\gamma} e^{\rho E_{ij,l}(k)}$ converts into a composite entropy function. From our experiments, we get the best results with $\gamma = 0.32$ and $\rho = 0.5$ on keystroke dataset. We compute $h(l)$ for all i, j and the minimum value associated with l gives the identity of the unknown user; so the criterion function is selected as:

$$H = \min\{h(l)\} \tag{34}$$

4. Description of Databases Used

For the evaluation of the keystroke dynamics based authentication system, the following publicly available datasets are availed:

***a*) CMU Keystroke Dynamics Benchmark Dataset [1]**

This database comprising 51 users is collected in 8 sessions and 50 repetitions of the same password are recorded in each session. We have 400 samples per user. CMU benchmark dataset has keystroke features, viz., DD (Down-Down) time, UD (Up-Down) time and H(Hold) time. A 10 character password (.tie5Roanl) is typed by a user. In our study, we have used H and UD since they give the best results. Accordingly, we have 21 features that include: 11 H values for 10 characters and an enter key, 10 UD values of time latencies between 11 key presses. Considering each of 51 users as both genuine and imposter we have a pool of 51x50 sets of experiments.

Half of feature vectors of every user in each session is treated as the training data and the remaining half as the positive test data, *i.e.* 200 samples each. In addition to this, the first 5 samples from each of the remaining users are assumed to be the negative test data in every experiment. As demonstrated in [29] that by including the background user's data during the training phase in keystroke dynamics, the error rates are reduced significantly. Similarly, to train a classifier, we take the first 4 samples of the remaining users as negative training data resulting in 196 samples. The classifier is trained in each experiment such that samples of an imposter are not visible to the classifier during training.

The authentication accuracy is evaluated using EER (Equal Error Rate) where False Acceptance Rate (FAR) equals False Rejection Rate (FRR) on ROC curve. FAR is the rate at which an unauthorized person (*i.e.* imposter) would be given access to the system as a genuine user [30] whereas FRR is the rate at which an authorized user would be rejected the access to the system considering him as imposter. FAR is calculated as the ratio of imposters granted access to the total number of imposter attempts while FRR is calculated as the ratio of genuine users denied access to the total number of genuine attempts.

For SVM and Random Forest Classifier, the performance measure EER is calculated for each set of genuine and imposter users, *i.e.* 51 × 50 sets of such experiments. The mean of the performance measure values (EERs) is then calculated for all the experiments. In addition to EER we also report FAR and FRR values and authentication accuracy.

For convex entropy based classifier, the performance is reported in terms of

EER(mean), FAR, FRR and Accuracy, calculated as the ratio of number of users correctly classified as genuine/imposter user to the total number of user attempts across all the experiments. FAR is calculated as the ratio of number of users which are incorrectly accepted as genuine to the total number of imposter user attempts across all the experiments. FRR is calculated as the ratio of number of users which are incorrectly rejected as imposter to the total number of genuine user attempts across all the experiments.

***b*) Sapientia University Keystroke Benchmark Dataset for Android platform [31]**

This data is collected from 42 users with 51 samples per user with at least 2 sessions per user. Each user types the password ".tie5Roanl" on Android based Mobile Devices Nexus 7 Tablet and Mobil LG Optimus L7 II P710. The key sequence resulting from typing the password is "ti e [123?] 5 [abc] [Shift] R [Shift] o a n l" which are 14 key presses. We have used all of 71 features of the dataset given in Table 1 for our work.

For every user 45 samples are randomly selected as the training data and the remaining 6 samples constitute the positive test data. We take 1 sample from each one of the remaining users so as to have 41 negative test samples and include the first 2 samples of the remaining users who are neither genuine nor imposter in the training data resulting in 80 samples for the negative class as the imposter training data. Here again, each one of 42 users is considered as both genuine and imposter to conduct 42×41 sets of experiments. The classifier is trained on each of these experiments such that the samples of an imposter are unavailable to the classifier during training. Performance is then measured in terms of error rates EER, FAR, FRR and accuracy.

***c*) Classification of the proposed features**

In our work, we have used three classifiers. The first is two-class SVM classifier with a linear kernel. The second classifier used is Random Forest Classifier, which generates an ensemble of decision trees based on the training data. Every test input vector is evaluated by all decision trees in the Random forest classifier

Table 1. Features Present in Android based SU dataset [31].

Feature Name	No. of Features
Key Hold Time (H)	14
Down-Down Time (DD)	13
Up-Down Time (UD)	13
Key Press Pressure (P)	14
Finger Area (FA)	14
Mean Hold Time	1
Mean Finger Area	1
Mean Pressure	1
Total	**71**

that operates on the principle of majority votes to get the classification vote. In addition to these standard classifiers the proposed convex entropy based classifier is third one discussed in Section 3.2.

5. Results of Implementation

Before presenting our results, let us see the state of the art on keystroke dynamics in the literature. Table 2 shows EERs for some of the algorithms with the best performance on the recent CMU dataset. The first algorithm given in Table 2 is an anomaly detector that uses Manhattan Distance [1] [32]. This method arrives at the mean of timing samples and the absolute mean standard deviation for each feature [32]. Given a test feature vector, a distance score is calculated using the following scaled Manhattan Distance:

$$\sum_{i=1}^{p} \frac{|x_i - y_i|}{a_i} \tag{35}$$

where x_i and y_i are ith test feature and ith mean vectors respectively and a_i is the mean absolute standard deviation of ith feature.

Zhong *et al.* [33] have developed the new distance metric by combining both Mahalanobis and Manhattan distances as given by

$$\|x-y\|' = \left\|S^{-1/2}(x-y)\right\|_1 \tag{36}$$

where $S^{-1/2}$ is the inverse of the principal square root of covariance matrix S.

Deng and Zhong [29] have used Deep Belief Networks by stacking together Gaussian RBM (Restricted Boltzmann Machine) with 31 visible units and 100 hidden units, and a binary RBM with 100 visible units and 100 hidden units, and obtained a mean EER of 0.035.

Table 3 shows the EERs obtained on SU dataset in [31] for two-class classifiers using all 71 features as shown in Table 1. These features also include touch based features such as finger area and key press pressure.

The performance of different information set based features with $\alpha = 1$ on CMU dataset is listed in Table 4 in terms of FAR, FRR, EER and accuracy using

Table 2. EER for different algorithms on CMU dataset.

Algorithm	EER
Manhattan(scaled) [1]	0.096
Combined Mahalanobis and Manhattan distance [33]	0.084
DBN [29]	0.035

Table 3. EER for SU dataset using the classifiers used in [31].

Classifier	EER
Random Forest	3.1%
Bayes Network Classifier	4.3%
K-NN	8.3%

Table 4. Comparison of results for various Information Set based features on CMU dataset with $\alpha = 1$ and SVM.

Features	FAR	FRR	EER (mean)	Accuracy (mean)
Information Feature	0.0157	0.0275	0.0201	0.9791
Sigmoid Feature	0.0156	0.0277	0.0201	0.9790
Energy Feature	0.0193	0.0326	0.0237	0.9748
Multi-Quadratic Feature	0.0391	0.0292	0.0334	0.9653
Hanman Transform	0.0291	0.0293	0.0290	0.9708
Complement Hanman Transform	0.0220	0.0235	0.0223	0.9773

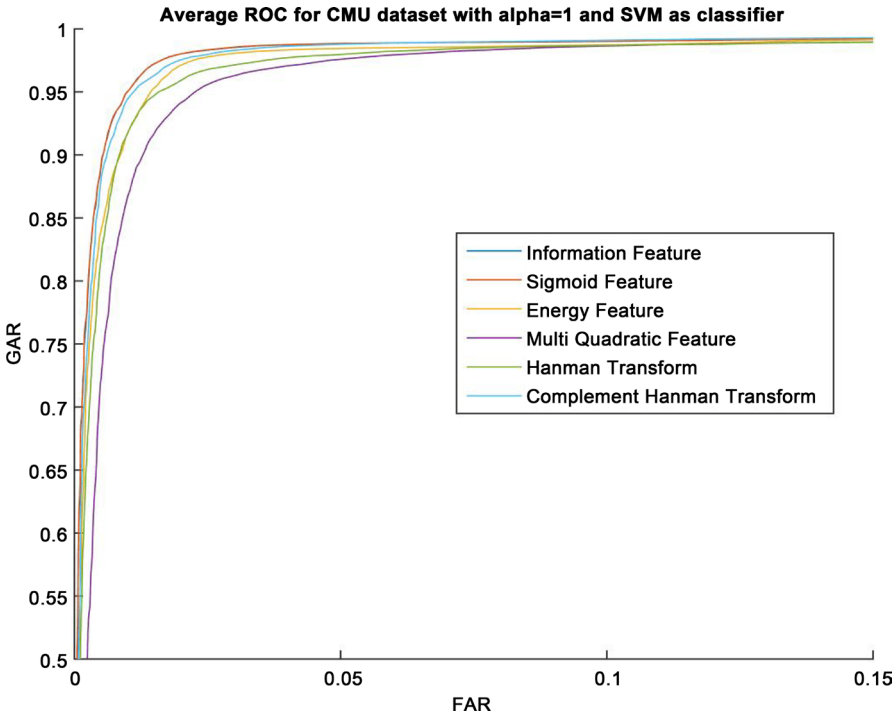

Figure 2. Average ROC for various Information Set based features on CMU dataset with $\alpha = 1$ and SVM as classifier.

SVM. The best EER of 0.0201 is obtained with Information and Sigmoid features. The average ROC for various information set features with $\alpha = 1$ on CMU dataset is shown in Figure 2 using SVM as classifier.

The features of Table 4 are applied on the same CMU dataset but with $\alpha = 2$ using SVM and the results are given in Table 5. Here the best EER of 0.0225 is obtained with the Sigmoid Features. Comparing Table 4 and Table 5 we note that sigmoid feature is best in terms of EER values. The average ROC for various information set features with $\alpha = 2$ on CMU dataset is shown in Figure 3 using SVM.

Table 6 shows the performance of Convex Entropy Based Classifier for different information set features with $\alpha = 1$ in terms of FAR, FRR, EER and Accu-

racy. Table 6 shows the best performance for Information Feature in terms of EER of 0.0112 and accuracy of 0.9875. The average ROC for various information set features with $\alpha = 1$ is shown in Figure 4. The features used in Table 6 are obtained with $\alpha = 2$ for Convex Entropy Based Classifier and the results are shown in Table 7. Here we get the best performance in terms of EER of 0.0111 and accuracy of 0.9866 for Composite Transform. The average ROC with $\alpha = 2$ is shown in Figure 5.

Table 5. Comparison of results for various Information Set based features on CMU with $\alpha = 2$ and SVM.

Features	FAR	FRR	EER (mean)	Accuracy (mean)
Information Feature	0.0193	0.0308	0.0227	0.9756
Sigmoid Feature	0.0191	0.0305	0.0225	0.9758
Energy Feature	0.0224	0.0379	0.0282	0.9707
Complement Hanman Transform	0.0357	0.0300	0.0319	0.9668

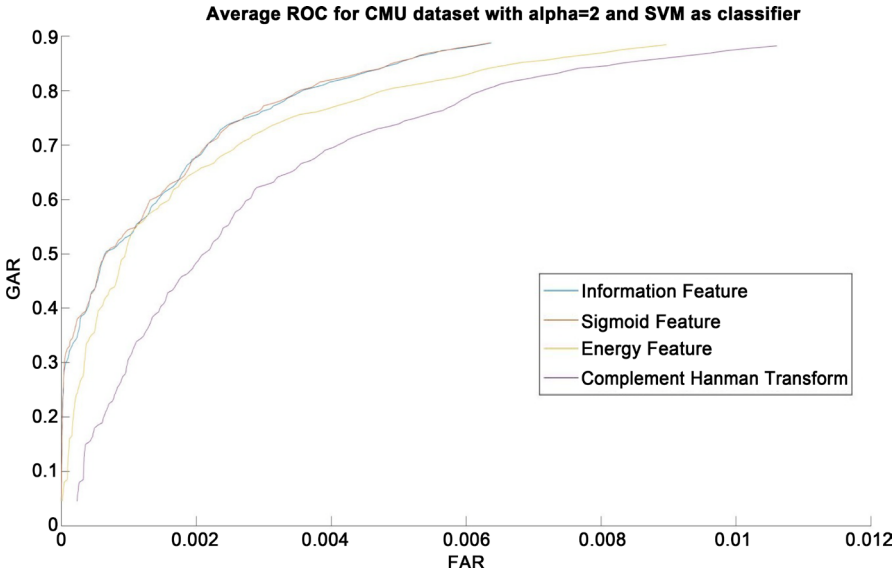

Figure 3. Average ROC for various Information Set based features on CMU dataset with $\alpha = 2$ and SVM as classifier.

Table 6. Comparison of results for various Information Set based features on CMU with $\alpha = 1$ and Convex Entropy Classifier.

Features	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0147	0.0099	0.0112	0.9875
Sigmoid Feature	0.0147	0.0099	0.0112	0.9874
Energy Feature	0.0126	0.0156	0.0127	0.9861
Complement Hanman Transform	0.0171	0.0081	0.0113	0.9869
Composite Transform	0.0180	0.0097	0.0125	0.9857
Complement Composite Transform	0.0198	0.0091	0.0129	0.9850

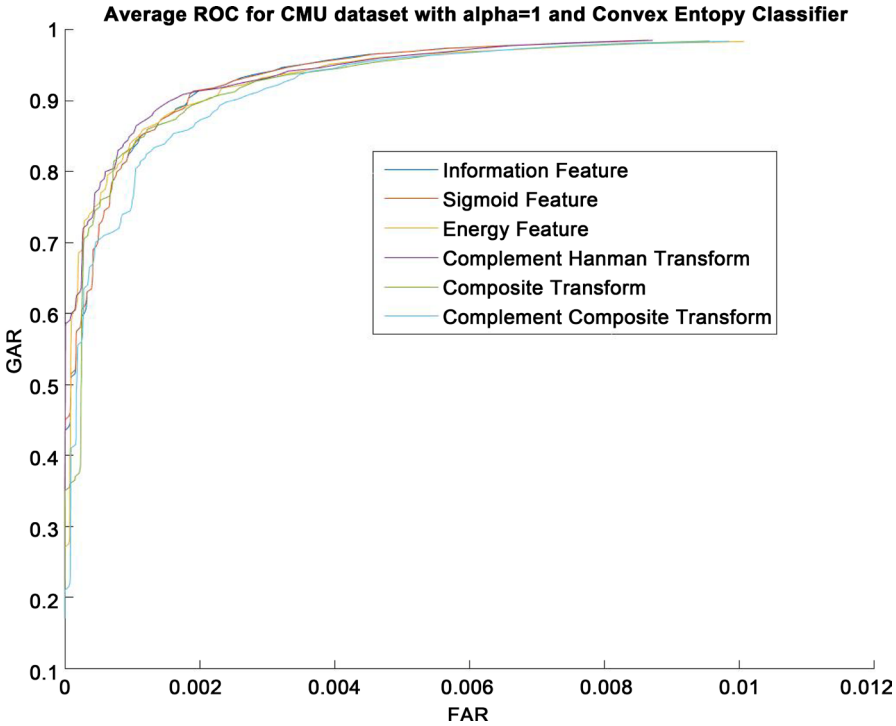

Figure 4. Average ROC for various Information Set based features on CMU dataset with $\alpha = 1$ and Convex Entropy Classifier.

Table 7. Comparison of results for various Information Set based features on CMU with $\alpha = 2$ and Convex Entropy Classifier.

Features	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0219	0.0124	0.0158	0.9823
Sigmoid Feature	0.0218	0.0122	0.0159	0.9825
Energy Feature	0.0191	0.0163	0.0168	0.9821
Complement Hanman Transform	0.0322	0.0181	0.0232	0.9741
Composite Transform	0.0177	0.0080	0.0111	0.9866
Complement Composite Transform	0.0181	0.0081	0.0118	0.9863

The features used in Table 4 and Table 5 along with the additional features contribute to EERs and accuracy figures in Table 8 on the same CMU data with $\alpha = 1$ but with random forest classifier. In this case the best EER of 0.0103 is obtained with the Hanman Transform. The averages of ROCs for some of the features are shown in Figure 6.

The features shown in Table 8 are now obtained with $\alpha = 2$. Random Forest is used on CMU dataset and the results are given in Table 9. The mean ROC curves for some of these features are displayed in Figure 7.

The results of some of the features of Table 8 and Table 9 used on SU dataset with Random Forest for $\alpha = 1$ are given in Table 10. By this classifier, the best EER is obtained with Information Value and Energy features. The Composite Transform lags these features in performance slightly. The average ROC for

these features is shown in Figure 8.

These features are also tested with $\alpha = 2$ on SU dataset with Random Forest

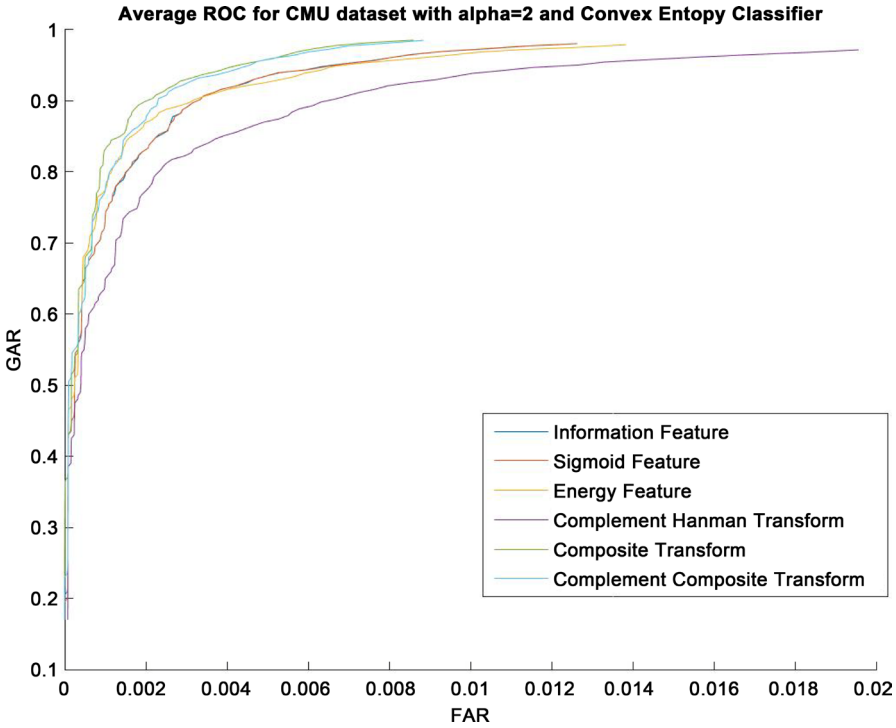

Figure 5. Average ROC for various Information Set based features on CMU dataset with $\alpha = 2$ and Convex Entropy Classifier.

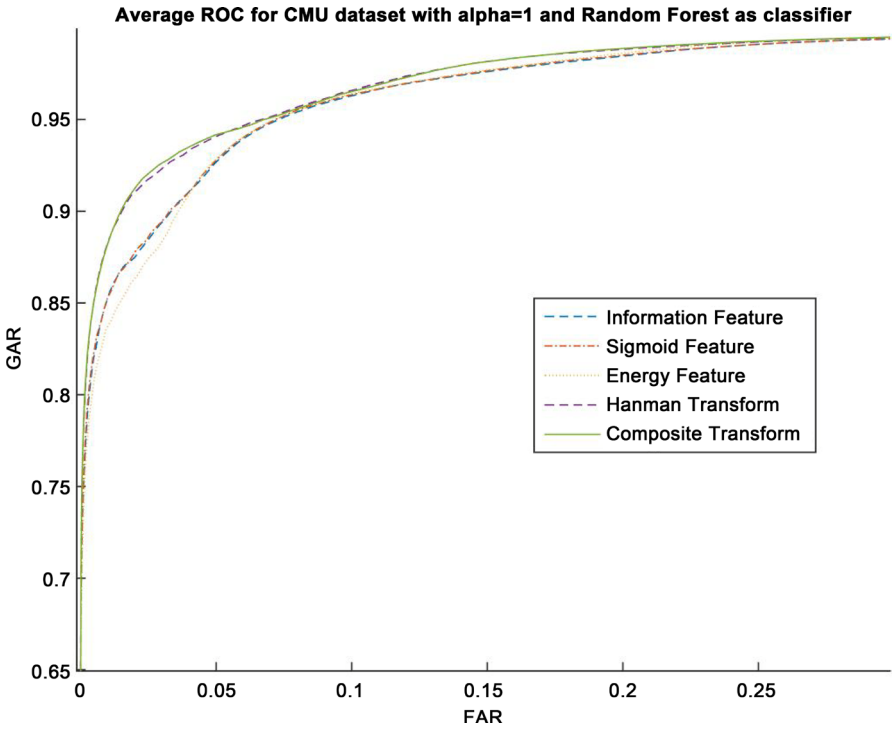

Figure 6. Average ROC for various Information Set based features on CMU dataset with $\alpha = 1$ and Random Forest as classifier.

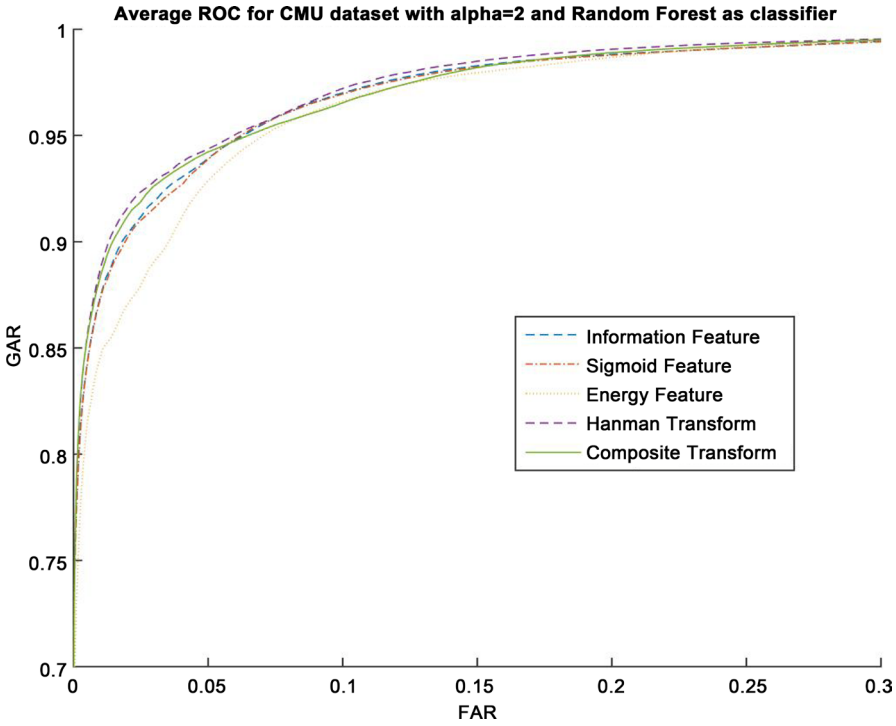

Figure 7. Average ROC for various Information Set based features on CMU dataset with $\alpha = 2$ and Random Forest as classifier.

Table 8. Comparison of results for various Information Set based features on CMU with α=1 and Random Forest (Treebagger) as classifier.

Feature	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0077	0.0251	0.0129	0.9846
Sigmoid Feature	0.0077	0.0252	0.0128	0.9845
Energy Feature	0.0084	0.0260	0.0131	0.9838
Hanman Transform	0.0082	0.0173	0.0103	0.9878
Multi Quadratic Feature	0.0114	0.0221	0.0146	0.9838
Inverse Multi Quadratic Feature	0.0128	0.0255	0.0169	0.9815
Complement Energy Feature	0.0128	0.0240	0.0160	0.9822
Complement Hanman Transform	0.0086	0.0183	0.0110	0.9871
Complement Information	0.0139	0.0267	0.0176	0.9804
CompositeTransform	0.0085	0.0168	0.0104	0.9878

Table 9. Comparison of results for various Information Set based features on CMU with α =2 and Random Forest (Treebagger) as classifier.

Feature	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0081	0.0242	0.0125	0.9848
Sigmoid Feature	0.0081	0.0244	0.0126	0.9847
Energy Feature	0.0078	0.0265	0.0130	0.9839
Hanman Transform	0.0090	0.0178	0.0109	0.9871

Continued

Multi Quadratic Feature	0.0112	0.0213	0.0145	0.9843
Inverse Multi Quadratic Feature	0.0109	0.0221	0.0148	0.9841
Complement Energy Feature	0.0155	0.0264	0.0180	0.9797
Complement Hanman Transform	0.0092	0.0177	0.0112	0.9870
Complement Information Feature	0.0168	0.0312	0.0210	0.9768
Composite Transform	0.0084	0.0167	0.0102	0.9879

Table 10. Comparison of results for various Information Sets based features on SU dataset with $\alpha = 1$ and Random Forest (Treebagger) as classifier.

Feature	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0216	0.0605	0.0228	0.9734
Sigmoid Feature	0.0331	0.0554	0.0286	0.9641
Energy Feature	0.0179	0.0676	0.0228	0.9757
Hanman Transform	0.0279	0.0556	0.0248	0.9686
Shannon Transform	0.0246	0.0600	0.0262	0.9708
Composite Transform	0.0279	0.0552	0.0268	0.9686

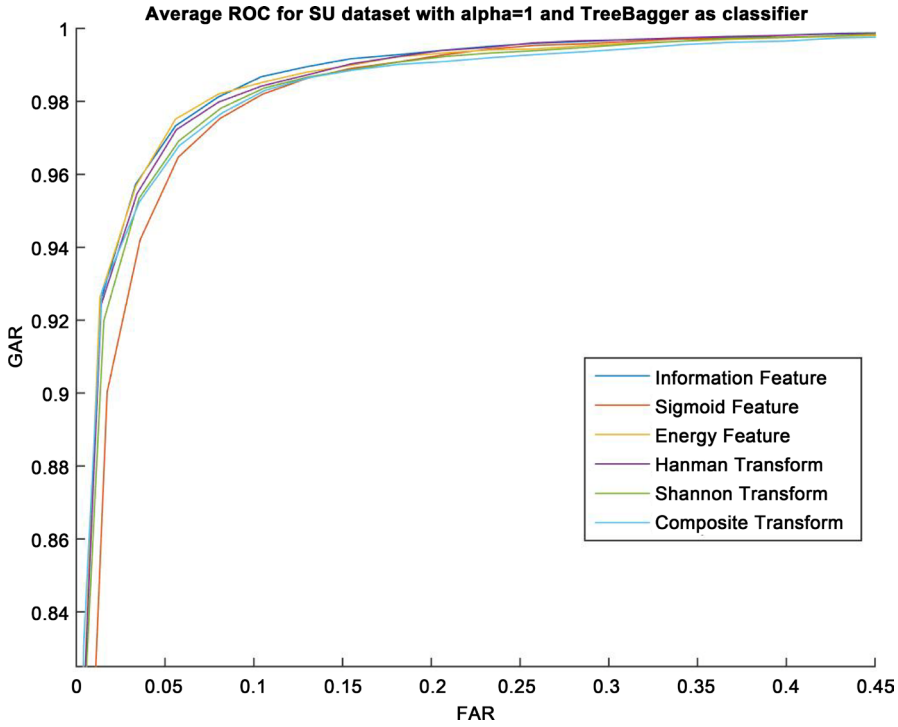

Figure 8. Average ROC for various Information Set based features on SU dataset with $\alpha = 1$ and Treebagger as classifier.

classifier and the results are shown in Table 11. The mean ROC for these features is shown in Figure 9. Best EER is obtained using Hanman Transform. Note that SVM and *Convex* Entropy classifier don't perform well because of very

Table 11. Comparison of results for various Information Sets based features on SU dataset with $\alpha = 2$ and Random Forest (Treebagger) as classifier.

Feature	FAR	FRR	EER (mean)	Accuracy
Information Feature	0.0225	0.0589	0.0236	0.9728
Sigmoid Feature	0.0488	0.0652	0.0446	0.9491
Energy Feature	0.0198	0.0574	0.0223	0.9754
Hanman Transform	0.0281	0.0569	0.0219	0.9682
Shannon Transform	0.0244	0.0518	0.0227	0.9721
Composite Transform	0.0270	0.0540	0.0249	0.9696

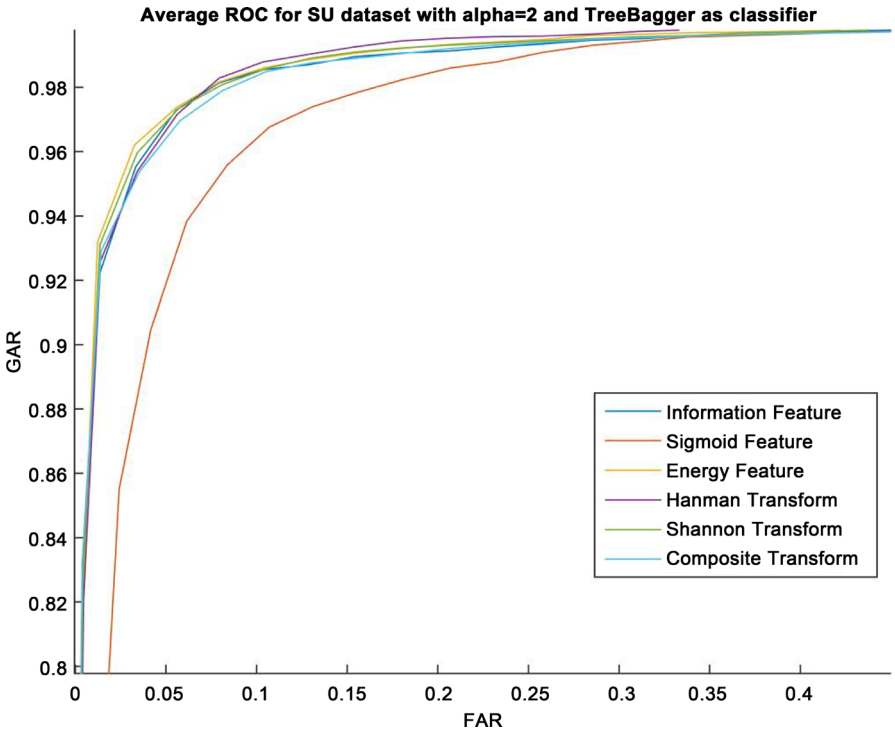

Figure 9. Average ROC for various Information Set based features on SU dataset with $\alpha = 2$ and Random forest also called Treebagger as classifier.

less data.

Discussion of Results: Out of 10 features, a subset of 6 features has been found to be effective on implementing two datasets: CMU and SU using three classifiers: SVM, Convex Entropy and Random Forest (Treebagger). The best results on CMU dataset are due to Composite Transform feature with EER of 0.0102 for Treebagger classifier and EER of 0.0111 for Convex Entropy classifier. The EERs obtained by the literature features (See Table 2) are inferior. On SU data, however Treebagger gives better result (EER of 0.228) with Energy feature than those of literature features in Table 3.

6. Conclusions

The possibilistic uncertainty in the keystroke timing values termed as informa-

tion source values when represented by the entropy function gives rise to the information values which are shown to be the products of information source values and the corresponding membership function values. Two Gaussian membership functions are employed: one using the mean and variance of all the samples which lead to temporal information values and the other using the mean and variance of a single sample which lead to spatial information values. These two kinds of information values, viz., spatial and temporal components are concatenated to provide us the two-component information set (TCIS) features. From the concatenated features, various new features such as Information Value, Energy, Sigmoid, Hanman Transform, Shannon Transform, Multi-quadratic, Composite Transform and their complements are generated. In this work, Hanman Classifier is redesigned by the use of Convex Entropy Function.

TCIS features from two benchmark datasets CMU and SU are classified using Convex Entropy based classifier, SVM and Random Forest classifiers. Their performance is evaluated on the proposed features in terms of error rates (FAR, FRR, EER) and accuracy. These features are also tested on Android Touchscreen based Mobile Keystroke Dataset and the performance of these features outperforms that of the literature features.

We plan to extend this work by considering new features based on information set theory and type-2 and interval fuzzy sets. It is observed that the efficiency of feature type is dependent on database. Out of all the features investigated, Sigmoid, Energy, Hanman transform and Composite transform features have made their mark as the effective features. We have not attempted the fusion of the effective features. If these features are fused by either at the feature level or score level, then the fused feature vector is likely to outperform on all the datasets considered.

There are two limitations of the proposed approach. The first limitation is that it is not suitable for capturing global characteristics as its main forte is in local characteristics. The second limitation is the choice of membership function. Generally Gaussian function serves as an effective membership function.

References

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

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

[3] 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* 12*th International Conference on Neural Information Processing*.

[4] Loy, C.C., Lai, W.K. and Lim, C.P. (2007) Keystroke Patterns Classification Using the ARTMAP-FD Neural Network. 3*rd International Conference on Intelligent In-*

formation Hiding and Multimedia Signal Processing, **1**, 61-64. https://doi.org/10.1109/IIH-MSP.2007.218

[5] Montalvão Filho, J.R. and Freire, E.O. (2006) *Pattern Recognition Letters*, **27**, 1440-1446.

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

[7] Gaines, R.S., Lisowski, W., Press, S.J. and Shapiro, N. (1980) Authentication by Keystroke Timing: Some Preliminary Results. No. RAND-R-2526-NSF.

[8] Young, J.R. and Hammon, R.W. (1989) Method and Apparatus for Verifying an Individual's Identity. Google Patents.

[9] Garcia, J.D. (1986) Personal Identification Apparatus. Google Patents.

[10] Joyce, R. and Gupta, G. (1990) *Communications of the ACM*, **33**, 168-176. https://doi.org/10.1145/75577.75582

[11] Bleha, S., Slivinsky, C. and Hussien, B. (1990) *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 1217-1222. https://doi.org/10.1109/34.62613

[12] Brown, M. and Rogers, S.J. (1993) *International Journal of Man-Machine Studies*, **39**, 999-1014. https://doi.org/10.1006/imms.1993.1092

[13] Umphress, D. and Williams, G. (1985) *International Journal of Man-Machine Studies*, **23**, 263-273.

[14] The, P.S., Teoh, A.B.J., Ong, T.S. and Neo, H.F. (2007) Statistical Fusion Approach on Keystroke Dynamics. 3*rd International IEEE Conference on Signal-Image Technologies and Internet-Based System*, 918-923. https://doi.org/10.1109/SITIS.2007.46

[15] Giroux, S., Wachowiak-Smolikova, R. and Wachowiak, M.P. (2009) Keystroke-Based Authentication by Key Press Intervals as a Complementary Behavioral Biometric. *IEEE International Conference on Systems, Man and Cybernetics*, 80-85.

[16] Bleha, S.A. and Obaidat, M.S. (1993) *IEEE Transactions on Systems, Man, and Cybernetics*, **23**, 900-902. https://doi.org/10.1109/21.256563

[17] Sang, Y., Shen, H. and Fan, P. (2004) Novel Impostors Detection in Keystroke Dynamics by Support Vector Machine. In: *Parallel and Distributed Computing: Applications and Technologies*, Springer, Berlin, 666-669. https://doi.org/10.1007/978-3-540-30501-9_128

[18] Pavaday, N. and Soyjaudah, K.M.S. (2007) Performance of the K nearest Neighbor in Keyboard Dynamic Authentication. *Proceedings of the* 2007 *Computer Science and IT Education Conference*, 599-604.

[19] Obaidat, M.S. (1995) A Verification Methodology for Computer Systems Users. *Proceedings of the* 1995 *ACM Symposium on Applied Computing*, 258-262.

[20] Zadeh, L.A. (1965) *Information and Control*, **8**, 338-353.

[21] Aggarwal, M. and Hanmandlu, M. (2016) *IEEE Transactions on Fuzzy Systems*, **24**, 1-15. https://doi.org/10.1109/TFUZZ.2015.2417593

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

[23] Mamta and Hanmandlu, M. (2013) *Expert Systems with Applications*, **40**, 6478-6490.

[24] Mamta and Hanmandlu, M. (2014) *Engineering Applications of Artificial Intelligence*, **36**, 269-286.

[25] Sayeed, F. and Hanmandlu, M. (2017) *Knowledge and Information Systems*, **52**,

485-507. https://doi.org/10.1007/s10115-016-1017-x

[26] Arora, P., Hanmandlu, M. and Srivastava, S. (2015) *Pattern Recognition Letters*, **68**, 336-342.

[27] Grover, J. and Hanmandlu, M. (2015) *Applied Soft Computing*, **31**, 1-13.

[28] Zhao, Y.-B., Fang, S.-C. and Li, D. (2006) *SIAM Journal on Optimization*, **17**, 37-51. https://doi.org/10.1137/040603838

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

[30] Crawford, H. (2010) Keystroke Dynamics: Characteristics and Opportunities. 2010 8*th International Conference on Privacy*, *Security and Trust*, 205-212. https://doi.org/10.1109/PST.2010.5593258

[31] Antal, M. and Szabó, L.Z. (2015) An Evaluation of One-Class and Two-Class Classification Algorithms for Keystroke Dynamics Authentication on Mobile Devices. 20*th International Conference on Control Systems and Computer Science*, 343-350. https://doi.org/10.1109/CSCS.2015.16

[32] Araújo, L.C.F., Sucupira, L.H.R., Lizarraga, M.G., Ling, L.L. and Yabu-Uti, J.B.T. (2005) *IEEE Transactions on Signal Processing*, **53**, 851-855. https://doi.org/10.1109/TSP.2004.839903

[33] Zhong, Y., Deng, Y. and Jain, A.K. (2012) Keystroke Dynamics for User Authentication. 2012 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 117-123. https://doi.org/10.1109/CVPRW.2012.6239225

Appendix A: Adaptive Mamta-Hanman Entropy Function

Let us make H adaptive in (3) by considering the parameters as functions rather than constants as per the original definition in [24]. For simplicity, we consider the adaptive form of 1D H given by

$$H_i = \sum_i p_i^{\alpha} \mathrm{e}^{-\left(cp_i^{\gamma}+d\right)^{\beta}} \tag{A.1}$$

In this the parameters c_i and d_i are assumed to be variables. The 1D form of this function [24] is:

$$H = \sum_{i=1}^{n} p_i^{\alpha} I\left(p_i\right) \tag{A.2}$$

where $I\left(p_i\right) = \mathrm{e}^{-\left(c_i p_i^{\alpha}+d_i\right)^{\beta}}$ with $c_i, d_i \in [0,1]$. We will prove some important properties of the adaptive entropy function. In order to simplify proofs, we set $\alpha = 1$.

Properties of Adaptive Entropy Function

1) $I\left(p_i\right) = \mathrm{e}^{-(c_i p_i + d_i)^{\beta}}$ is a continuous function for $\forall p_i \in [0,1]$; so $p_i \mathrm{e}^{-(c_i p_i + d_i)^{\beta}}$ is also a continuous function being a product of two continuous functions and H being the sum of continuous functions is also a continuous function.

2) $I\left(p_i\right)$ is bounded. As $\mathrm{e}^{-(c_i p_i + d_i)^{\beta}} < 1$, $p_i^{\alpha} \mathrm{e}^{-(c_i p_i + d_i)^{\beta}}$ is bounded for $\forall i$; so is H bounded.

3) With the increase in p_i, $I\left(p_i\right)$ decreases; so $\dfrac{\partial I\left(p_i\right)}{\partial p_i} = -c_i \beta \left(c_i p_i + d_i\right)^{\beta-1} \mathrm{e}^{-(c_i p_i + d_i)^{\beta}} < 0$; as $c, \beta > 0$.

4) If $p_1 = p_2 = p_3 = \cdots = p_n = \dfrac{1}{n}$ then H is an increasing function of n.

$$H = \sum_{i=1}^{n} \frac{1}{n^{\alpha}} \mathrm{e}^{-\left(\frac{c_i}{n}+d_i\right)^{\beta}} = \frac{1}{n^{\alpha-1}} \mathrm{e}^{-\left(\frac{c_i}{n}+d_i\right)^{\beta}} \tag{A.3}$$

$$\frac{\partial H}{\partial n} = \frac{1}{n^{\alpha}} \mathrm{e}^{-\left(\frac{c_i}{n}+d_i\right)^{\beta}} \left[(1-\alpha) + \frac{c_i \beta}{n} \left(\frac{c_i}{n} + d_i \right)^{\beta-1} \right] > 0 \tag{A.4}$$

Hence this is proved.

5) Note that $H = \sum_{i=1}^{n} p_i^{\alpha} \mathrm{e}^{-(c_i p_i + d_i)^{\beta}}$ is a concave function where $p_i \in [0,1]$ and $\sum_{i=1}^{n} p_i^{\alpha} = 1$.

To prove that this is concave the Hessian matrix must be negative definite. The Hessian is computed as follows:

$$\frac{\partial H}{\partial p_i} = p_i^{\alpha-1} \mathrm{e}^{-(c_i p_i + d_i)^{\beta} \left[\alpha - \beta c_i p_i (c_i p_i + d_i)^{\beta-1} \right]} \tag{A.5}$$

$$\frac{\partial^2 H}{\partial p_i^2} = \mathrm{e}^{-(c_i p_i + d_i)^{\beta} \left[\alpha(\alpha-1) p_i^{\alpha-2} - \beta c_i p_i (c_i p_i + d_i)^{\beta-1} \left((\alpha-1) + \alpha p_i^{\alpha-2} \right) + \beta c_i^2 p_i^{\alpha-1} (c_i p_i + d_i)^{\beta-2} \left(\beta p_i (c_i p_i + d_i)^{\beta} - \beta p_i + 1 \right) \right]} \tag{A.6}$$

As c_i, p_i are in [0, 1].

$$p_i = \frac{1}{n},\ \frac{\partial^2 H}{\partial p_i^2} = c_i\left(\frac{c_i}{n} - 2\right) < 0 \tag{A.7}$$

$$\frac{\partial^2 H}{\partial p_i \partial p_j} = 0 \quad \text{and} \quad H_F = \begin{bmatrix} \beta_1 & 0 & \cdots & 0 \\ 0 & \beta_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \beta_n \end{bmatrix} \tag{A.8}$$

where $\beta_i = c_i\left(c_i p_i - 2\right) < 0$, hence all the Eigen values of the Hessian matrix are negative. So, the Hessian is negative definite and H is concave.

6) Entropy H is maximum when all p_i's are equal. In other words, $p_i = \frac{1}{n}, \forall i$

That is,

$$\left(p_1, p_2, \cdots, p_n\right) = \left(\frac{1}{n}, \frac{1}{n}, \cdots, \frac{1}{n}\right) \tag{A.9}$$

In that case,

$$\beta_i = c_i\left(\frac{c_i}{n} - 2\right) < 0, \forall i \tag{A.10}$$

7) The entropy is minimum if and only if all p_i's except 1 are equal to zeros and single $p_i = 1$.

To make better representation of uncertainty, we will introduce higher form of uncertainty representation.