Abstract—The use of computer has increased rapidly as well as the use of internet applications such as e-commerce, online banking services, webmail, and blogs. All internet applications require a password authentication scheme to make sure only the genuine individual can login to the application. Passwords and personal identification numbers (PIN) have traditionally been used to access such applications [1, 2, 3]. However, it is easy for unauthorized persons to access these systems without detection. This paper addresses the issue of enhancing such systems using keystroke biometrics as a translucent level of user authentication. The paper focuses on using the time interval (key down-down) between keystrokes as a feature of individuals’ typing patterns to recognize authentic users and reject imposters. A Multilayer Perceptron (MLP) neural network with a Back Propagation (BP) learning algorithm is used to train and validate the features.

Keywords - Keystroke, Biometrics, Multilayer Perceptron (MLP) Neural Network, Back Propagation (BP), Verification

I. INTRODUCTION

Keystroke dynamics is one of the novel and creative biometric techniques. It is not only nonintrusive, but also transparent and inexpensive [3]. Some research has been published [4] that shows authentication via typing biometrics reported since 1990. Some aspects can be utilised to create a keystroke verification system, for example by using a target string that will be typed by the user and monitored by the system, or a number of samples collected during the enrolment process to compound the training set. Two widely used features are duration of the key time interval that key remains pressed, and keystroke latency time interval between successive keystrokes. A more robust way is to use a combination of these features to analyse a keystroke system. Time accuracy, trials of authentication and classifiers are other examples of ways to analyse a keystroke authentication system [3, 5, 6, 7].

Statistical models and diagraph latencies were found to be the first techniques used to analyse keystroke biometrics. Then the neural network (NN) approach was developed by [8], they used a simple MLP with Back Propagation (BP). Their work was extended by [9], who considered the deviation on the architecture and parameters of the neural network with customised keystroke latency and gave a 1.1% False Acceptance Rate (FAR) and 0% impostor pass rate (IPR).

N. Capuano [10] used the MLP with Radial Basis Function (RBF) as a transfer function, rather than a sigmoid one used previously by others. It resulted in 97% correct authentication with 0% intrusions. Work in [11] achieved 97.5% correct classification by using a combination of multilayer feedforward with BP algorithm (MFN/BP) and sum of product (SOP) network with keystroke time interval. M.S Obaidat continued this work with S Sadoun and used key hold times for classification, comparing the performance with the former interkey time based technique and then combining interkey and hold times for the identification process. An identification accuracy of 100% was achieved when hold and interkey times were combined and trained using fuzzy ARTMAP, RBF and learning vector quantization (LVQ) neural networks [4].

In [12], keystroke was classified based on an MLP approach and K-means cluster algorithm. Both the MLP and K-means gave an 84% and 85% acceptance rate and a 69% and 85% impostor rejection rate. Alternatively, [13] designed and developed a system that combined the maximum pressure applied on the keyboard and time latency between keystrokes as features to create typing patterns for each user. They combined an Artificial Neural Network (ANN) with a MFN and an Adaptive Neuro-Fuzzy Interference System (ANFIS) as classifiers to authenticate individual users. The classification rate achieved was 100% with an average training time of 0.9094 sec. However, in the work of A. Sulong, Wahyudi and M.U. Siddiqi only a combination of the maximum pressure applied on the keyboard and time latency between keystrokes were combined as features to create typing patterns for each user using a RBFN [14]. The 100% classification rate with 22.4 sec in average training time achieved shows that the RBFN-based authentication system is suitable for keystroke analysis functions. In this study we investigate a MLP neural network using a BP algorithm as a classifier for keystroke biometrics authentication system.

II. DATABASES

In this paper, all databases used are based on the work of [15], which can be accessed from the Biochaves site [16]. It consists of three databases (A, B, and C) with down-down (DD) time intervals only.
In Database A, 10 people were asked to type a set of four words, *chocolate, zebra, banana, taxi*, 10 times; 5 times (five samples) during the first session and then another 5 times (five samples) a month later. Database B was built up in a similar way to Database A, but only 8 people took part. Also, the duration between the first and second sessions was shorter, being only a week [15].

In Database C, the DD time intervals were recorded by typing two fixed words in Portuguese, *computador calcula*. There were 14 people involved in this database. They were given copies of the sampling program and were free to type the words when and where they liked. In all databases, if the subjects pressed ‘Delete’ or ‘Backspace’ then they had to retype the string from the beginning in order to reduce the chance of recording poor samples and categorized as static text.

Each databases divided into two datasets, one for training dataset that contains 5 samples from each person, and the other 5 samples for test datasets.

III. PREPROCESSING
This work used two types of preprocessing technique to the databases before it can be analyse.

A. Equalization Histogram
All the databases were normalized using an equalization histogram which is a nonlinear transformation. A straightforward equalization transform obtain using following equation:

$$q(x) = \frac{1}{1+\exp\left(-\frac{K(\log_2(x) - \mu_2)}{\sigma_2}\right)}$$  \hspace{1cm} (1)

where $K = 1.7$, $\mu_2 = -1.56$, $\sigma_2 = 0.65$ (estimated from Databases A, B and C) and $y = \log_2(x)$ with $x$ is given in seconds. This work used $q(x)$ as input to the MLP neural network.

This simple nonlinear memoryless mapping of time intervals used as it can significantly improve the performance of authentication algorithms. It transformed the unbalanced probability density function (pdf) to the balanced pdf of the random variable that models the time intervals and increases the performance of most algorithms. The value of $K$, $\mu_i$ and $\sigma_i$ are estimated from Databases A, B and C.

B. Principal Component Analysis (PCA)
Each input data should be preprocessed so that its mean value is close to zero, or else it will be small compared to its standard deviation. Furthermore the input data should be uncorrelated and this can be done using PCA. PCA, also known as the Karhunen–Loève transformation in communication theory, maximizes the rate of decrease variance, doesn’t have any distributional assumption, and it can reduce the dimensionality of vectors without losing valuable information.

Principal component analysis is based on the statistical representation of a random variable. Suppose we have a random vector population $x$, where

$$x = (x_1, ..., x_n)^T$$  \hspace{1cm} (2)

and the mean of that population is denoted by

$$\mu_x = E[x]$$  \hspace{1cm} (3)

and the covariance matrix of the same data set is

$$C_x = E[(x-\mu_x)(x-\mu_x)^T]$$  \hspace{1cm} (4)

The components of $C_x$, denoted by $c_{ij}$, represent the covariance between the random variable components $x_i$ and $x_j$. The component $c_{ij}$ is the variance of the component $x_i$. The variance of a component indicates the spread of the component values around its mean value. If two components $x_i$ and $x_j$ of the data are uncorrelated, their covariance is zero ($c_{ij} = c_{ji} = 0$). The covariance matrix is, by definition, always symmetric.

From a symmetric matrix such as the covariance matrix, we can calculate an orthogonal basis by finding its eigenvalues and eigenvectors. The eigenvectors $e_i$ and the corresponding eigenvalues $\lambda_i$ are the solutions of the equation

$$C_x e_i = \lambda_i e_i, \quad i = 1, ..., n$$  \hspace{1cm} (5)

From here let $A$ be a matrix consisting of eigenvectors of the covariance matrix as the row vectors. By transforming a data vector $x$, we get

$$y = Ax - \mu_x$$  \hspace{1cm} (6)

which is a point in the orthogonal coordinate system defined by the eigenvectors. We can reconstruct the original data vector $x$ from $y$ by

$$x = A^T y + \mu_x$$  \hspace{1cm} (7)

using the property of an orthogonal matrix $A^{-1} = A^T$.

IV. MULTILAYER PERCEPTRON NEURAL NETWORK (MLP NN)
MLP is a feed forward neural network pattern that maps groups of input data onto a set of target outputs. Figure 1 shows the structure of the MLP network used in this paper. It consists of three main parts: an input layer, one hidden layer, and an output layer.
The input layer distributes the input data to the processing elements in the next layer. The second stage is the hidden layer which incorporates the nonlinearity behaviour and the last stage shows the output layer. Input and output are directly accessible, while the hidden layers and adaptive learning rate backpropagation is a network where the nonlinearity function used in this work that updates weights and bias is the non-linear activation function (Hyperbolic tangent sigmoid transfer function (tansig)). The output of neuron $j$ in the hidden layer is given by:

$$H_j = f \left( \sum_{i=1}^{n} w_{ij} x_i + b_i \right)$$

(8)

where $w_{ij}$ and $b_i$ are the hidden layer neurons weight and bias, and $f(.)$ is the non-linear activation function (Hyperbolic tangent sigmoid transfer function (tansig)). Then the output of the network is:

$$y = f \left( \sum_{j=1}^{m} w_{kj} H_j + b_o \right)$$

(9)

where $f(.)$, $w_{kj}$ and $b_o$ are the output layer neuron activation function (again Hyperbolic tangent sigmoid transfer function (tansig) was used), weights and bias respectively. The BP algorithm was chosen to minimize the mean square error (MSE) based on the set of $N$ given a training data pattern as following equation [13]:

$$E = \frac{1}{2} \sum_{n=1}^{N} (d_n - y_n)^2$$

(10)

where $d$ is referred as the target or desired output and $y_n$ is neural network output.

The training set is repeatedly presented to the network until the output of the neural network, $y_n$, is steady and close to the target, $d$. Gradient descent with momentum and adaptive learning rate backpropagation is a network function used in this work that updates weights and bias. The weights are updated to get a minimum $E$. During the simulations the number of input nodes, learning rate value, the number of hidden nodes, momentum value and performance goal value were changed so as to find the most suitable parameter values.

Figure 1. Architecture and signal flow of an MLP neural network

The output of neuron $j$ in the hidden layer is given by:

$$H_j = f \left( \sum_{i=1}^{n} w_{ij} x_i + b_i \right)$$

(8)

where $w_{ij}$ and $b_i$ are the hidden layer neurons weight and bias, and $f(.)$ is the non-linear activation function (Hyperbolic tangent sigmoid transfer function (tansig)). Then the output of the network is:

$$y = f \left( \sum_{j=1}^{m} w_{kj} H_j + b_o \right)$$

(9)

where $f(.)$, $w_{kj}$ and $b_o$ are the output layer neuron activation function (again Hyperbolic tangent sigmoid transfer function (tansig) was used), weights and bias respectively. The BP algorithm was chosen to minimize the mean square error (MSE) based on the set of $N$ given a training data pattern as following equation [13]:

$$E = \frac{1}{2} \sum_{n=1}^{N} (d_n - y_n)^2$$

(10)

where $d$ is referred as the target or desired output and $y_n$ is neural network output.

The training set is repeatedly presented to the network until the output of the neural network, $y_n$, is steady and close to the target, $d$. Gradient descent with momentum and adaptive learning rate backpropagation is a network function used in this work that updates weights and bias. The weights are updated to get a minimum $E$. During the simulations the number of input nodes, learning rate value, the number of hidden nodes, momentum value and performance goal value were changed so as to find the most suitable parameter values.

V. RESULT AND DISCUSSION

Figure 2 shows correct acceptance rate for keystroke biometrics authentication system using MLP NN and two different preprocessing techniques. Both techniques show promising result with more than 80% correct classified valid users.

MLP neural network is layered feedforward networks that produce nonlinear function mappings by hidden and output layers of MLP are both nonlinear. This makes MLP more suitable and highly accurate to classifier non linear keystroke data.

Also the nodes in the hidden and output layers of MLP use the same or monotonic activation functions where it computes inner products from the input and the incoming weights; thus the activation of a hidden unit in a MLP is constant on results hyperplanes surfaces. A MLP also forms a distribution representation where many hidden units will typically contribute to the determination of the output values make it more accurate than linear classifier such as Euclidean distance.

Furthermore, by using equalization histogram technique, as nonlinear transformation to the databases, increased performance of MLP NN between 97% and 98% corrects acceptance rate. While by using PCA to the databases gives 80% to 89% performance rate. This result can be explained by the fact that PCA is a linear transformation technique which less efficient for nonlinear data. The features extracted by PCA are actually global features for all pattern classes, thus they are not necessarily much representative for discriminating one class from others.

Figure 2. Comparison performance of MLP using different preprocessing
This paper shows that keystroke is a special behavioural biometric that can be used as features for an additional and transparent layer of user authentication. This is demonstrable by using a MLP NN platform. The simulation results revealed that MLP with BP network is suitable to discriminate and classify a nonlinear keystroke database as high as 98% in correct acceptance rate. It also shows that MLP NN with nonlinear transformation shows greater promising result for improving CAR in order to verify the authorized user as compared to linear transformation technique. Moreover this work proves that MLP NN gives better accuracy and improvement in classifying keystroke nonlinear data. However, further study has to be done to improve the correct acceptance rate with different method of discriminant analysis and also enhance the level of security of the system.

ACKNOWLEDGMENT

N. Harun would like to thank Universiti Teknologi Malaysia (UTM) for scholarship of doctoral study and also for their supporting.

REFERENCES