User identification and classification using raw accelerometer data from walking activity

Umair Ahmed
Department of Electrical Engineering, UET Lahore
theumairahmed@gmail.com

***Abstract*—User personalization and authorization is one of the hottest areas of research as it has widespread applications in the development of security and user personalized applications. Many biometric markers have been used extensively in the past to classify users based upon the features extracted from these markers e.g. one of the most widely used biometric markers for security purposes is the human fingerprint, others worth mentioning are facial and voice features. In this project we propose the use of the human gait recorded through inertial measurement units such as accelerometer to classify users and investigate the gait pattern of different human users to investigate if it is a viable marker for classification and uniqueness due to inherent distinction in the natural walking pattern of human beings. We have used feature extraction algorithms to preprocess and extract features from raw accelerometer data followed by the training of various supervised learning classifiers and evaluating their performance.**

***Index Terms*— biometric authentication, gait classification, supervised learning, user personalization, wearable sensors**

I. Introduction

User authentication is one of the earliest problems when it comes to security. With the boom in the amount of information stored over computer networks and shared data storage banks, the area of information security has been very active to provide secure and authorized access to this information.

In the process of evaluating different biometric patterns for secure user authorization the human gait has been particularly found to be an interesting phenomenon in various researches where it has been recorded by means of image processing, by using sensors such as force sensor or by radar based signal processing. Studies such as [5], [6] have shown that the gait patterns indeed show a certain degree of distinction between different users

Inertial measurement units such as accelerometers and gyroscopes are now a ubiquitous part of our daily lives. Nearly every smartphone employs an accelerometer and a gyroscope for different applications. The purpose of our research was to record the data from these sensors from different users to study if we can distinguish amongst the user based upon this raw data.

II. Related Work

In the past a variety of works has been seen using the gait as a biometric marker for user recognition. One of the earliest work on gait analysis for classifications is done by [1] and [2] by means of video capture and analyzing the changes in the gait pattern over time using image processing techniques.

Wearable systems were also recently employed for the purpose of gait identification and classification. [3] Worked on the use of accelerometer to detect patterns in the human gait. [4] Extends this work by adding Gyroscope together with the Accelerometer to record the gait characteristics for this problem.

III. Dataset

The dataset employed for our project is taken from the online UCI Machine learning repository [7] consists of raw accelerometer data recorded from an Android smartphone positioned in the chest pocket of 22 participants. The participants were asked to walk for a certain time and the data was acquired for this walking activity in separate CSV files. Some of the CSV files contained data which contained either very small set of values or had enormous noise for the most part hence, we had to delete those files reducing the dataset to 18 participants out of 22.

The dataset initially consists of four attributes. The first one being the timestamp at which the sensor outputs the data and the next three being the x, y and z acceleration values recorded from the accelerometer. The problem with three separate axes values is that it is hard to visualize the walking pattern as we had to plot the three axes separately. Also due to noise and false readings the values of x, y and z axes cannot be compared on one scale.

One very easy approach to solve this problem is to calculate another axis called the Resultant vector from the magnitudes of individual acceleration axes i.e.

$$R = \sqrt{a_x^2 + a_y^2 + a_z^2}$$

Hence the final dataset consists of an additional axis which makes it very easy to visualize the acceleration pattern for different users. Figure 1 plots the Resultant vector axis for different users simplifying the visualization of the user data.

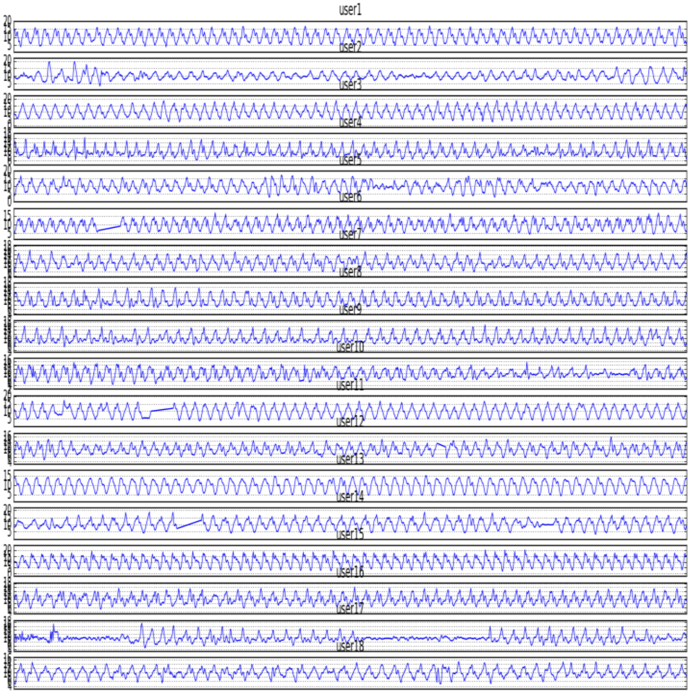

Figure 1: Plot of Resultant vector data for different users

It can be seen that there exists a certain degree of distinction between the walking patterns of different users as testified by the plot of the data. Our goal now remains to train a machine learning agent to distinguish between different users by looking at this raw data alone.

IV. Methods

The attributes present in our dataset are all continuous and as seen from the plot they do possess a certain periodic pattern. However for any machine learning classifier to classify this raw data into different users we have to train it using a training set of feature vectors. Hence the problem of user classification is formulated as follows:

1- Record the raw acceleration data from accelerometers for different users.

2- Preprocess the data and apply feature extraction to extract the set of feature vectors used for training the classifier.

3- Label the feature vectors with the IDs of the users.

4- Train the classifier using a portion of the extracted features.

5- Evaluate the performance of classifier on the remaining portion of the feature vectors.

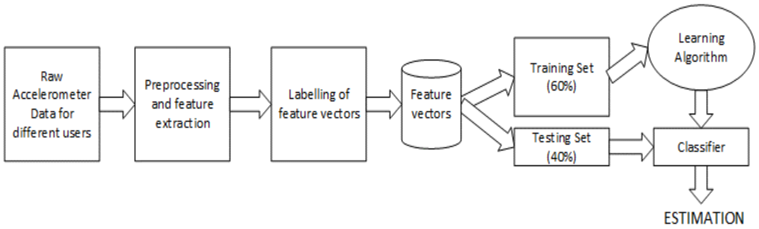
Raw Accelerometer Data for different users
Preprocessing and feature extraction
Labelling of feature vectors
Feature vectors
Training Set (60%)
Testing Set (40%)
Learning Algorithm
Classifier
ESTIMATION

Figure 2: Problem Formulation and flowchart of the solution

The choice of features to be extracted from raw data is important for the accuracy for classification. Ideally only those features should be extracted which provide the most insight about the human gait to the classifier. After a brief overview of previous work by [8] Table 1 shows the features extracted for our classification problem.

Feature	Description
mean	Mean
std	Standard deviation
var	Variance
min	Minimum value
max	Maximum value
acf_mean	Auto correlation mean
acf_std	Auto correlation standard deviation
acv_mean	Auto covariance mean
acv_std	Auto covariance standard deviation
skew	Skewness
kurtosis	Kurtosis value
error	Deviation from mean

Table 1: Features extracted from raw accelerometer data

The preprocessing of the raw data and labelling of the feature vectors was done using a script written in python which performed a windowing operation on the raw data. The above mentioned 12 features were extracted from each of the 4 axis hence giving us total 48 different attributes from the raw sensor data.

The size of the window was selected to contain 100 data points with 50% overlap to spread out the gait features between consecutive windows. Figure 3 shows the windowing operation performed on one of the axes.

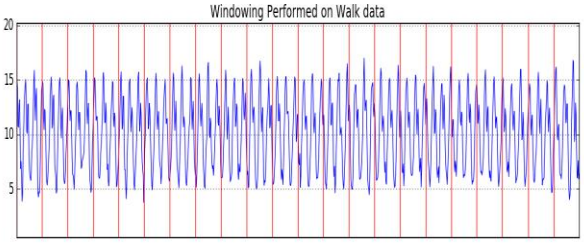

Figure 3: Windowing operation on raw data with 50% overlap

V. Experiments

The resulting labelled data was split into 60% for training and 40% for testing purposes. This section describes the performance of six different classifiers on the testing data and the confusion matrix for each showing the accuracy of classification:

1. Dummy Classifier:
For the baseline a Dummy classifier was chosen from the collection of classifiers in Python's scikit-learn library which uses a stratified strategy to generate predictions by respecting the training set's class distribution. The accuracy of classification was very poor being only **4.9%**. Figure 4 shows the confusion matrix for baseline classifier.

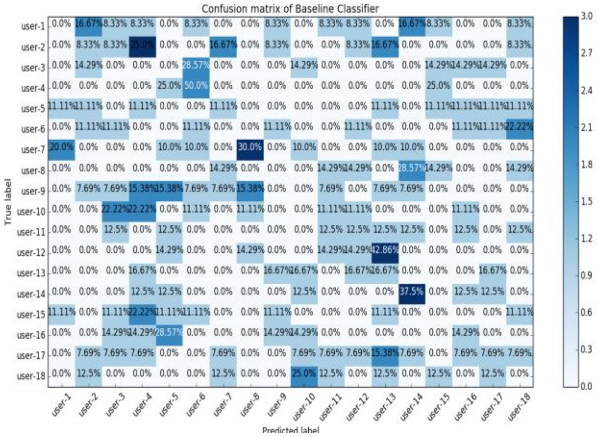

Figure 4: Confusion matrix for Baseline classifier (Prediction accuracy: 4.9%)

2. SVM (Kernel = 'RBF'):
The second classifier was a Support Vector Machine using a Radial Basis Function kernel. The SVM is a supervised learning classifier that classifies objects based on the support vectors of a dataset or points lie closest to the decision Boundary. SVM maximize the distance between support Vectors and the decision boundary [8]

The SVM trained on the training data predicted the testing data with an accuracy score of **27.5%** accuracy relatively better than the baseline score but still very poor.

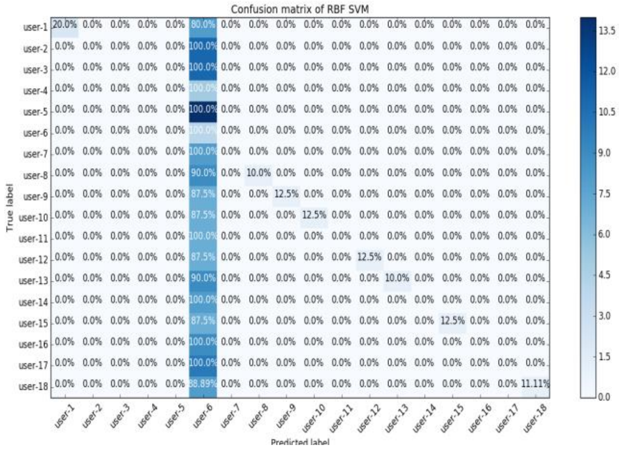

Figure 5: Confusion matrix for SVM (Prediction accuracy: 27.5%)

3. Decision Tree:
The next classifier experimented on the training set was a decision tree classifier with a specified maximum depth of six. It was observed that increasing the maximum depth of the tree increased the prediction accuracy but the classifier was prone to overfitting. To develop a general model of the data a depth of six was found optimal.

The Figure 6 shows the confusion matrix for the classification by the Decision Tree classifier resulting into an accuracy of **61.7%.**

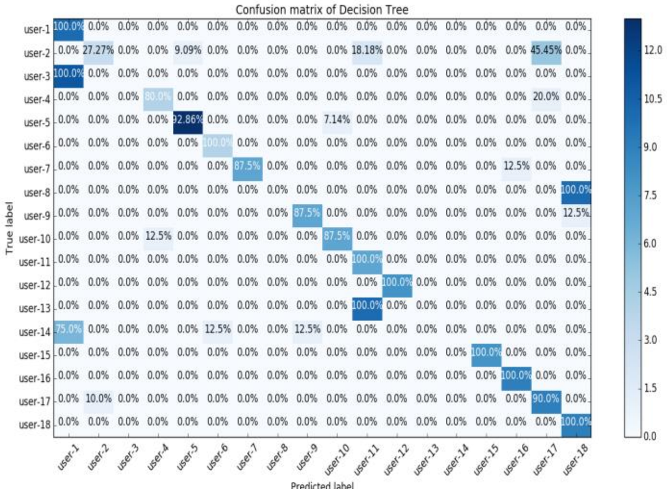

Figure 6: Confusion matrix for Decision Tree classifier (Prediction accuracy: 61.7%)

4. Random Forest Classifier:
A random forest is a Meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting [9].

The Random forest classifier selected for this project had a maximum allowable depth of 10 and the maximum features to split on equal to 1. The resultant accuracy of classification was found to be **83.2%**. The strongly highlighted diagonal of the confusion matrix in Figure 7 shows that most of the testing set was properly classified into the correct users.

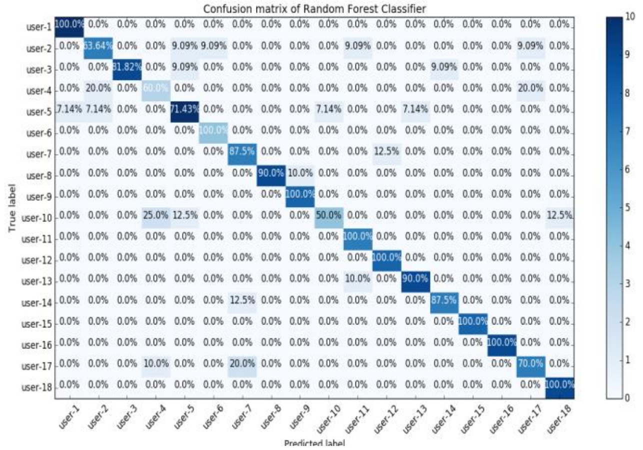

Figure 7: Confusion matrix for Random Forest Classifier (Prediction accuracy: 83.2%)

5. KNN:
The K-Nearest Neighbor algorithm estimates the classification of an unseen instance using the classification of the instance or instances that are closest to it, in some sense that we need to

define [10]. For our classification problem the value of K was selected to be 10 and the classifier predicted the users with an accuracy of **91.3%.** The figure below shows the confusion matrix drawn for KNN classifier.

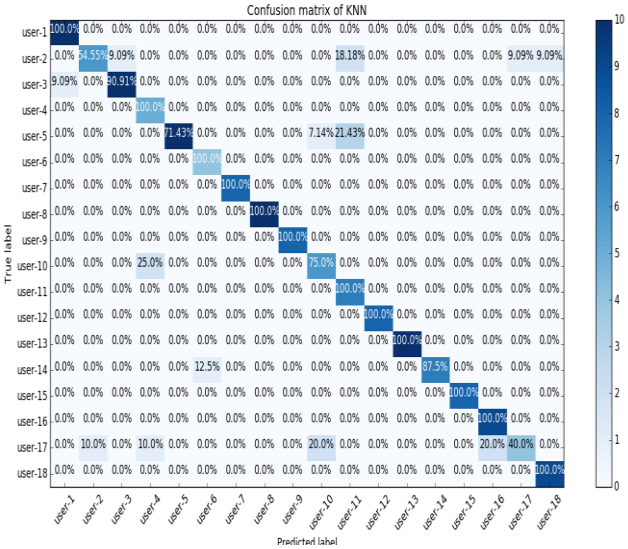

Figure 8: Confusion matrix for K-Nearest Neighbor classifier (Prediction accuracy: 91.3%)

6. SVM (Kernel = 'Linear'):

To further increase the accuracy of prediction the last classifier used was again a Support Vector Machine but this time with a Kernel of type Linear. With the kernel changed the accuracy of prediction further improved to a value of **94.1%** which is the best so far for each of the classifier tested. The confusion matrix also improved as the diagonal values increased showing that the classifier did not confuse users with each other and classified them correctly for most of the instances.

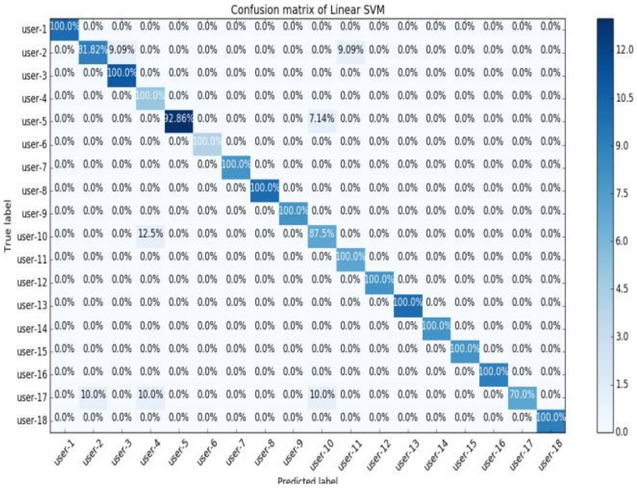

Figure 9: Confusion matrix for SVM with Linear Kernel (Prediction accuracy: 94.1%)

VI. RESULTS

The previous discussion shows the performance of different classifiers on the training set data. Figure 10 summarizes the performance of all of the six classifiers by showing the results of a 10-fold cross-validation tested on each classifier.

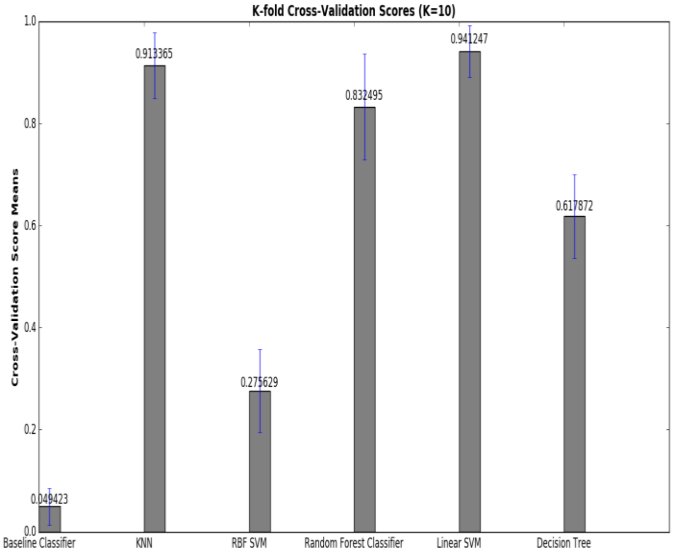

Figure 10: Summary of Cross validation scores for different classifiers

The gray bars show the mean of 10-fold cross-validation scores and the blue whiskers show the standard deviation of the scores.

In order to further evaluate the performance of this classifier on the training data we calculate some other metrics such as Precision, Recall and F-score which are defined by the following equations:

$$P = \frac{T_p}{T_p+F_p} \qquad R = \frac{T_p}{T_p+F_n} \qquad F = 2\,\frac{P\ X\ R}{P\ +\ R}$$

Where:

$T_p = True\ positives$

$F_p = False\ Positives$

$F_n = False\ negatives$

Figure 11 plots the values of these three metrics for the classification results of the Linear SVM to exhibit the performance of classifier for different classes of users.

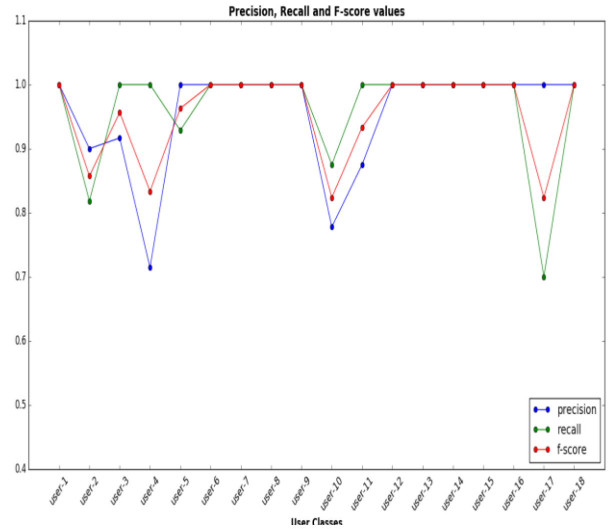

Figure 11: Precision, Recall and F-score plot of Linear SVM for different user classes

VII. DISCUSSION

Figure 10 shows the results of 10-fold cross-validation using the six classifiers demonstrating a baseline accuracy of 4.9%. The worst performance was observed using the SVM classifier with RBF Kernel giving a prediction score of 27.5% however the SVM classifier using the Linear Kernel was found out to be the best performing classifier with a top prediction score of 94.1%.

Figure 11 exhibits the performance of Linear SVM classifier towards the classification of different users. The low precision values for user-4 and user-10 show that the classifier performed comparatively poor in classifying these users correctly as it confused their gait pattern with others. This might be due to the relative generalness or high similarity of these user's gait patterns with other users.

VIII. CONCLUSION AND FUTURE WORK

In this paper we have worked on the pre-processing, feature extraction and classification methods on the problem of user recognition using raw smartphone sensor data. Our experiments have shown that gait can indeed be used as a viable biometric marker to solve the problem of user recognition and authentication.

In the future we would like to extend this problem by incorporating more datasets from more volunteers. Also instead of using just one smartphone accelerometer, the number of sensors attached to the body can be increased to detect more patterns in the natural human gait and perform better and refined classifications.

The complete code and dataset for the project can be found at: https://github.com/theumairahmed/User-Identification-and-Classification-From-Walking-Activity

REFERENCES

[1] Lily Lee and Eric Grimson. Gait analysis for recognition and classification. In *IEEE Conference on Face and Gesture Recognition*, pages 155–161, 2002.

[2] Lily Lee and Eric Grisom. Gait Appearance for Recognition. *Lecture Notes in Computer Science*, 2359:143–154, 2002.

[3] Casale, P. Pujol, O. and Radeva, P. Personalization and user verification in wearable systems using biometric walking patterns. *Personal and Ubiquitous Computing*, 16(5), 563-580, 2012.

[4] M. F. Nowlan, "Human Identification via Gait Recognition Using Accelerometer Gyro Forces," 2009.

[5] Nixon, Mark S and Carter, John N, "ON GAIT AS A BIOMETRIC: PROGRESS AND PROSPECTS," in *Proc. EUSIPCO*, Vienna, 2004.

[6] Jeffrey E. Boyd, James J. Little, "Biometric Gait Recognition," in Advanced Studies in Biometrics, Springer Berlin Heidelberg, 2005, pp. 19-42.

[7] P. Casale, "UCI Machine Learning Repository: User Identification From Walking Activity Data Set," 2012. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/User+Identification+From+Walking+Activity. [Accessed 12 November 2016].

[8] S. Z. Zhongyan Wu, "Human Activity Recognition using Wearable Devices Sensor Data," Palo Alto, 2015.

[9] Python scikit-learn, "scikit-learn 0.18.1 documentation," [Online]. Available: http://scikit learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html. [Accessed 13 December 2016].

[10] M. Bramer, Principles of Data Mining, Springer.