



Gait Identification using Smartphone Handheld Placement with Linear Interpolation Factor, Single Magnitude and One-vs-One Classifier Mapping

Abdul Rafiez Abdul Raziff^{1*} Md Nasir Sulaiman¹ Norwati Mustapha¹ Thinagaran Perumal¹

¹*Faculty of Computer Science and Information Technology,
Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia*

* Corresponding author's Email: abdrafz@gmail.com

Abstract: Gait signal of a person can be easily obtained using a smartphone sensor. To get the source of the signal, the smartphone need to be placed in the pocket, pouch or attached to other parts of the body. In the real world application, it is hard to place the device on the mentioned position. The easiest way is to put it on hand. In another issue, the single magnitude is known in the use of multiple orientations. However, this method may discard useful features for machine learning classification. Another problem is that the signal captured using a smartphone is not in a fix sampling rate and in the small distance, hence interpolation needs to be applied so that the sampling can be in a fix sequence with more fix point data. However, too much application of interpolation may result in low prediction rate. Finally, a multiclass dataset may contain overlapped class boundary which produces low accuracy on a single classifier mapping. In this paper, hand based smartphone placement position is implemented and evaluated. Single magnitude application is also evaluated in representing multiple positions of a person into one signal. Besides that, the linear interpolation factor is introduced in sampling the signal. Lastly, OvO classification model is implemented in binarizing the multiclass gait dataset. From the experiment, it shows that using the mentioned method do produce satisfactory result hence opening a new gateway in a better gait identification/recognition system.

Keywords: Gait identification, Handheld smartphone placement, Single signal magnitude, Linear interpolation factor, OvO classifier mapping

1. Introduction

Gait recognition has been used in many research and application. With the rapid development of microelectromechanical system (MEMS) technology, gait can be recorded easily without the need of expensive devices. The inertial sensor such as accelerometer and gyroscopes are part of the inertial measurement unit. Gait is a term that comes from the manner of walking (moving on foot) which consist of two steps. A normal walk may consist many gaits [1]. Every person may have their own style of the gait cycle, which is unique.

In the current gait research and application using the smartphone, most of the time the smartphone is placed in the pocket or in a pouch. In wearable sensor using a smartphone, the position varies among the past researcher. Most common positions

are in pocket [2-4], pouch [3-6], clipped to the waistband of the clothes[7] and multiple body position[8]. Based on the mentioned position and work, it can be seen that none of them tried placing the phone on the palm or handheld. Furthermore, in the real world situation, it is hard to get a gait signal of a person if that particular person does not have a pocket or pouch to be used as a placement of the smartphone. The best solution is to be placed on the hand. However, this poses a question whether signal captured from the smartphone on hand is usable for the identification process.

In other issue, single magnitude is used in overcoming the difference in the smartphone acceleration position in capturing accelerometer signal especially in the device's placement and orientation [2,9-10]. However, the viability of single magnitude in the handheld smartphone position is unknown as handheld position may produce

different signal and it may produce different type of noise if compared to other body position.

In capturing the gait signal using smartphone, there is a drawback due to the sensor API which only limit to the low sampling rate in reducing the battery power consumption. To overcome this situation, the obtained data need to be increased its sampling rate depending on the number of samples per second (Hz) which also called as interpolation. According to [11], there are generally 2 methods in implementing interpolation on the gait signal which is linear [12-18] and cubic spline [12]. There are also papers that do not mentioned the use of any interpolation in gait application [19-20]. Linear interpolation is a simple method of creating a new data points within a range of discrete data from a curve fitting using linear polynomials. For spline interpolation, new data points generated using piecewise polynomial but it produces smaller interpolation error than linear interpolation due to its capability in low degree polynomials for the spline [21]. However, the main issue in this application is the number of factor of linear interpolation that should be applied is not fixed in many signals based experiments.

Another issue that arise is the overlapped class boundary which may reduce the efficiency of the classifier [22]. In classification, multiclass classification has been well known in many types of research. One of the well-known methods is one-vs-one (OVO) classification. The rationale of using this method is to disseminate the overlapped classes boundaries. The idea of using this method is to transform the original multiclass problem into binary subsets using binarization method.

One of the work in comparing between OvO and OvR (One-vs-rest) has been conducted by [23] using a dataset from satellite image in classifying the built up area, vegetation and water. According to the author's judgment, OvO and OvR produce vary accuracy result based on the uniqueness of the dataset. However, in another work which also comparing the performance of OvO and OvR done by [24], the accuracy result and computational efficiency produced by OvO method is better than OvR. The dataset used was from many domains.

In the work conducted by [2,6,8,16] for gait recognition using accelerometer, single classifier method has been adopted for classification. The result is quite convincing but it can be further improved in creating a more efficient and robust learning structure.

The purpose of this paper is to investigate the accuracy performance on the proposed method

which on gait signal classification on different handheld smartphone position. The single magnitude is applied to evaluate whether it is applicable in the handheld based placement. Besides that, the linear interpolation factor is introduced and evaluated. The purpose of this method is to investigate whether using small walking distance is capable in the gait recognition. After that, the accuracy of the OvO classification layout on gait signal identification is evaluated. Then, multiple machine learning algorithms such as k-NN, MLP, SVM and J48 (decision tree) will be analyzed in finding the best classifier for OvO classification in this experimentation. All of the mentioned experiment will be evaluated through a single confusion matrix in determining the overall accuracy.

The main contributions of this paper are as follows,

- (i) The proven viability of handheld smartphone placement of smartphone for gait recognition.
- (ii) The single magnitude approach which is not usable in this work or application in representing multiple body placement of the smartphone as one based signal.
- (iii) Present an ideal linear interpolation factor for gait identification in generating new samples in a fixed length. At the same time, using this method is capable in identifying the signal in a small walking space.
- (iv) The application of the OvO multiclass method in gait identification incorporating with machine learning algorithm which is proven to increase the accuracy of the classification.
- (v) Comparison of multiple well-known machine learning classifier such as k-NN, MLP, SVM and J48 in OvO multiclass method.

The paper is organized as follows, Section 2 discuss about the proposed method, Experimental setup is discussed in Section 3, Implementation, results and discussion are discussed in Section 4 and finally Section 5 consist of conclusion and future works.

2. Proposed Method

2.1 Handheld Based of the Smartphone Accelerometer Sensor

In this paper, the handheld mobile phone position is examined in order to investigate whether the captured gait signal can be used for identification. Basically, there are three handheld

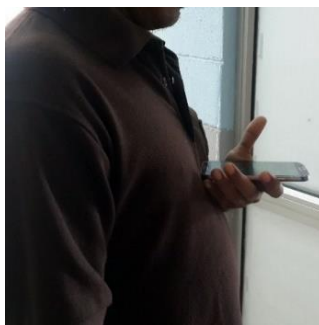


Figure. 1 Position of smartphone which is on hand held touching the upper abdomen (dataset 1)



Figure. 2 Position of smartphone which is on hand held (dataset 2)



Figure. 3 Position of smartphone which is on hand swing (dataset 3)

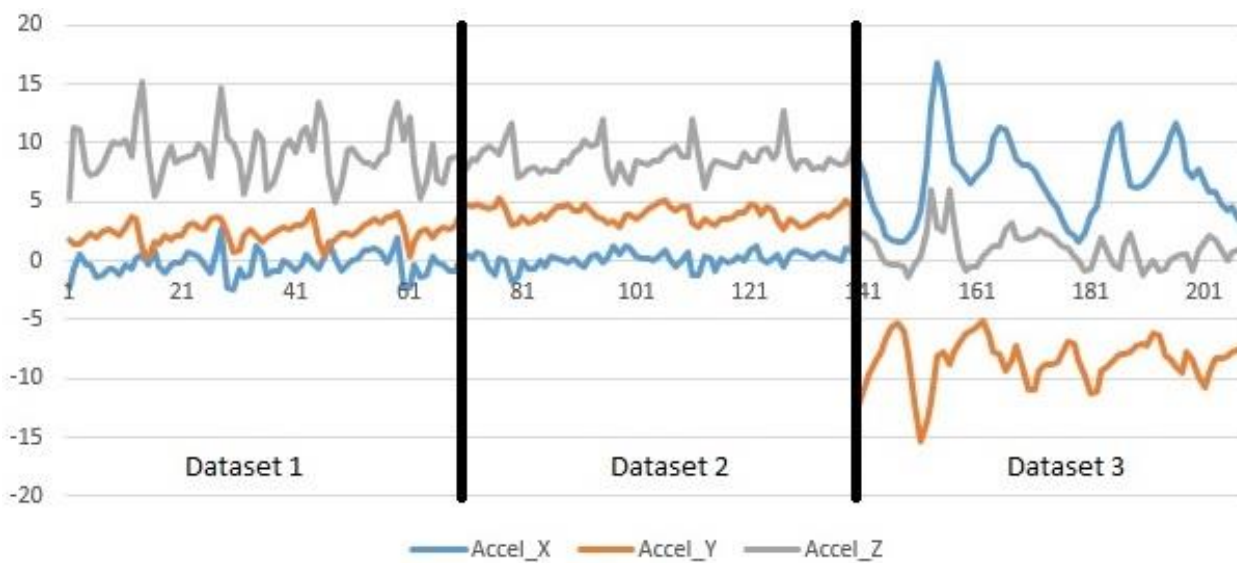


Figure. 4 The signal of a person while walking for dataset 1, dataset 2 and dataset 3

positions which are hold the phone while touching the upper abdomen (dataset 1), hold the phone forward of the abdomen while on hand palm (dataset 2) and on hand swing (dataset 3) as shown in Fig. 1, 2 and 3 respectively. The signal of a walking person for all the datasets are shown in Fig. 4. The measurement involved are acceleration on X, Y and Z axis which are horizontal movement, vertical movement and depth/height movement respectively.

From the hypothesis, it is assume that dataset 1 could produce the best result as the smartphone is considered as the most stable placement as its position is secured at the subject's abdomen whereas dataset 2 and dataset 3 are held independently which may result in inconsistent signal due to noise.

The viability of this position is measured from the score of the voting accuracy from the confusion matrix.

2.2 Single Magnitude

Single magnitude is a process of combining the three accelerations signals (x, y, z) into a single value which is the magnitude. The formula is as below:

$$r = \sqrt{x^2 + y^2 + z^2} \quad (1)$$

The purpose of converting the above into magnitude is to omit the difference in the acceleration position and placement of the mobile phone as previously done by [2,9-10].

The research question is how good this method could improves the accuracy although the number of usable axis is reduced from 3 to 1. The assumption is that it might lose the richness of the features before training and classification.

In this paper, the single magnitude is tested in the handheld based gait identification whether the result is satisfactory or not from the generated confusion matrix.

2.3 Single Magnitude

Accelerometer data recorded is based on *onSensorChanged* from Android API which the data is generated when the movement is triggered [6,16]. The sensor is not set to record the data at a fix sampling rate. To overcome this situation, linear interpolation has been applied at a determined factor to produce a sampling data over 1 second (Hz). In Table 1 shows the result based on the factor used for

applying this method. The formula of the used linear interpolation is as below:

$$s' = s_0 + \frac{(s_1 - s_0)(t' - t_0)}{t_1 - t_0} \quad (2)$$

where s_0 and s_1 represent two samples that are collected at time t_0 and t_1 respectively. s' and t' are representation of new sample and time that lies between (s_0, t_0) and (s_1, t_1) .

To represent a new sample at 50% between (s_0, t_0) and (s_1, t_1) , the easier formula is as below:

$$s' = \frac{s_0 + s_1}{2} \quad (3)$$

The research question in this method is how many of linear interpolation could be applied in-order to achieve an acceptable accuracy result. It is known that when this method is applied, the number of dataset would be increased. Hypothetically, with the increase of the dataset, it may affect the overall accuracy result which the accuracy may increase or decrease depending on the number of the application.

$$\text{linear interpolation}_n = \sum_{i=1}^n s' \quad (4)$$

The factor of the linear interpolation represents the number on how many linear interpolation should be applied to a dataset. The accuracy in this experiment depends on the last stage that will determine which factor would be the best to be applied.

2.4 One-vs-one (OvO) Multiclass Classification Model

In this paper, one-vs-one classification model is used where all dataset of classes are paired to generate new learned models according to the formula

$$\text{generated_learned_model}_n = \frac{n(n-1)}{2} \quad (5)$$

where n is the number of classes. In this method, we created n number of binary learned model for classification from the paired training data. The rationale of using this method is to eliminate the overlapped class boundary as can be seen in Fig. 5 hence preventing or reducing the probability of the estimated class to be mistakenly classified to other classes. It is also known that it is easier to classify binary class rather than multiclass dataset in machine learning.

Based on the Eq. (5), the number of generated paired dataset depends on the number of classes. So, if a dataset consists of 5 classes, the number of

generated paired dataset will be 10 as can be seen in Fig. 6.

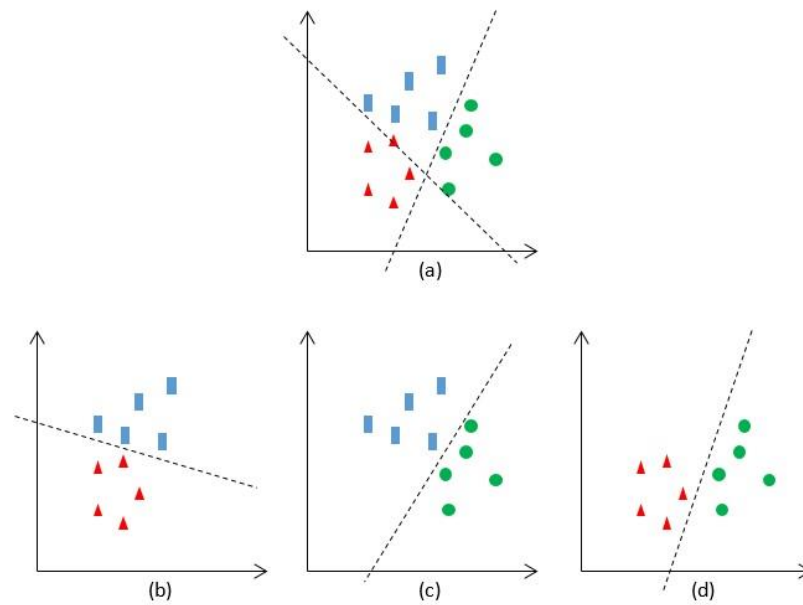


Figure. 5 Data distribution: (a) Sample of 3 classes, (b) (c) (d) after conversion into binary

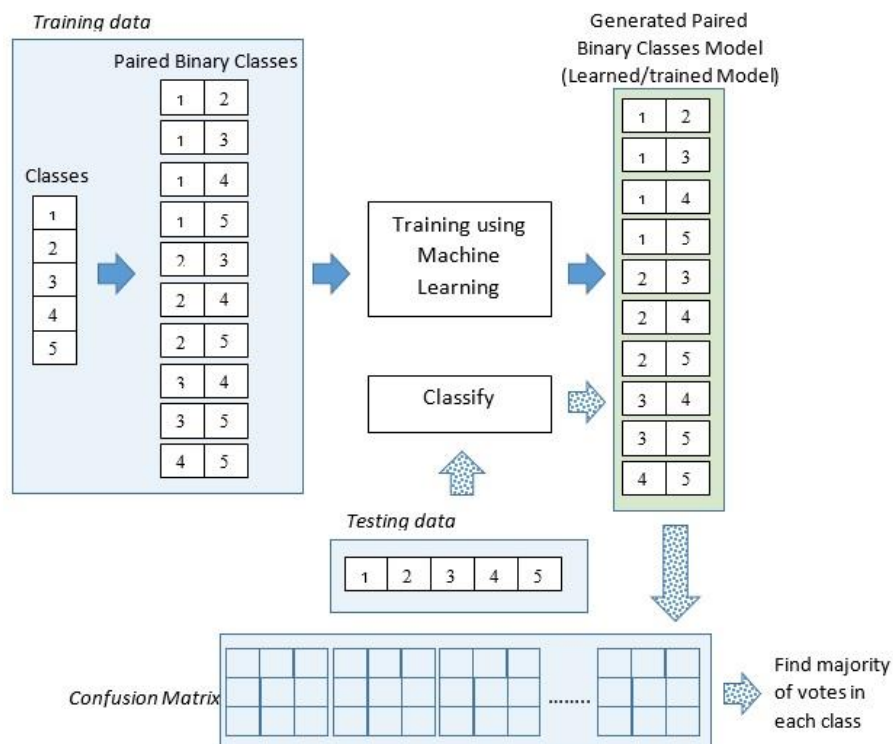


Figure. 6 Sample of 5 classes in arrangement of OvO multiclass mapping

Pairing of classes starts from the training dataset. The original dataset that contains multiclass label needs to be sorted according to the class label. Using looping, the dataset will be in the paired position. For example for paired data with class 1 and 2, only dataset that falls between class 1 and 2 only will be selected and will be assigned into a new paired dataset.

During training, machine learning algorithm will be applied for each of the generated paired training dataset. So, when the training is completed, the generated learned models will be categorized based on the paired train dataset.

For the testing phase or classification, decision is made by aggregating the decisions of the learned binary model by using simple majority voting where each binary classifier votes for predicted class [25-26]. The class with the maximum number of votes is predicted as an aggregated result of the classification from the generated confusion matrix from the motivation:

$$f(x) = \mathit{arg} \max_i (\sum_j f_{ij}(x)) \quad (6)$$

where f_{ij} is the classifier where class i are positive examples and class j are negative examples.

3. Experimental Setup

The experiment is divided into multiple levels as shown in Fig. 7.

3.1 Data Collection

Data is collected while walking on the straight line where the distance is approximately 10 meters. The mobile phone is held by hand instead of locating the phone in the pocket or in a pouch. This is because, in real world situation, a person may not carry a pouch or even has a pocket to put the phone. The best solution is to hold it on hand.

There are 30 subjects (person), which 15 of them are male and the rest are female. The age group is between 23-35 years old. The subject needs to walk on normal pace for about 15 meters for three different poses for the training set. On the second day, the subject needs to walk again for three different poses for the testing set.

Although the number of the classes is not many, however it is hope that this work might open a foundation in a more complex experimentation using the proposed framework model. In the work

conducted by Ren, [8], the number of subjects (classes) is 26. In other work by Hoang [16], the number of subject involved is even smaller which is only 14. Somehow, it is assumed that this number of classes would be adequate for a small trial.

The additional dataset is obtained from [5] to act as a control. The purpose is to prove the viability of the proposed method especially that are mentioned in section 2.3 and 2.4. The dataset consists of 408 people (classes) that were recorded using smartphone's accelerometer.

3.2 Preprocessing

The pre-processing methods involve 1) single signal magnitude (Section 2.2); 2) linear interpolation (Section 2.3); 3) centering around zero; and 4) fixed size overlapping sliding window.

3.3 Feature Extractions

The methods for features extraction involved are 1) minimum and maximum value; 2) mean; 3) standard deviation, 4) correlation, 5) root mean square; 6) signal vector magnitude; 7) number of zero crossing of the median; and 8) percentile rank.

3.4 Learning and Classification

Machine learning algorithm will be integrated with the OvO model (Section 2.4) to generate new learning model. Comparison of classifiers is performed with the application of k-NN, MLP, SVM and J48. Other machine learning algorithm has been tested but does not yield good result.

4. Experimental Result and Discussion

In this experiment, for dataset 1, 2 and 3, a total of 180 samples were used which 90 samples are divided for training and the balance for the testing phase. For each training and testing phase, there are 3 different phone locations which are on palm touching the abdomen (dataset 1), on hand palm while walking (dataset 2) and on hand swing (dataset 3) as described in methodology.

For dataset 4 which is taken from [5], there are 816 samples that are divided into training and testing set in equal size which are 408 each.

Single magnitude has been performed on the raw acceleration signal, combining all three axis into one. The result of its application is shown in Table 1.

Linear interpolation has been tested for each factor, from factor 1 to factor 4. An experiment has

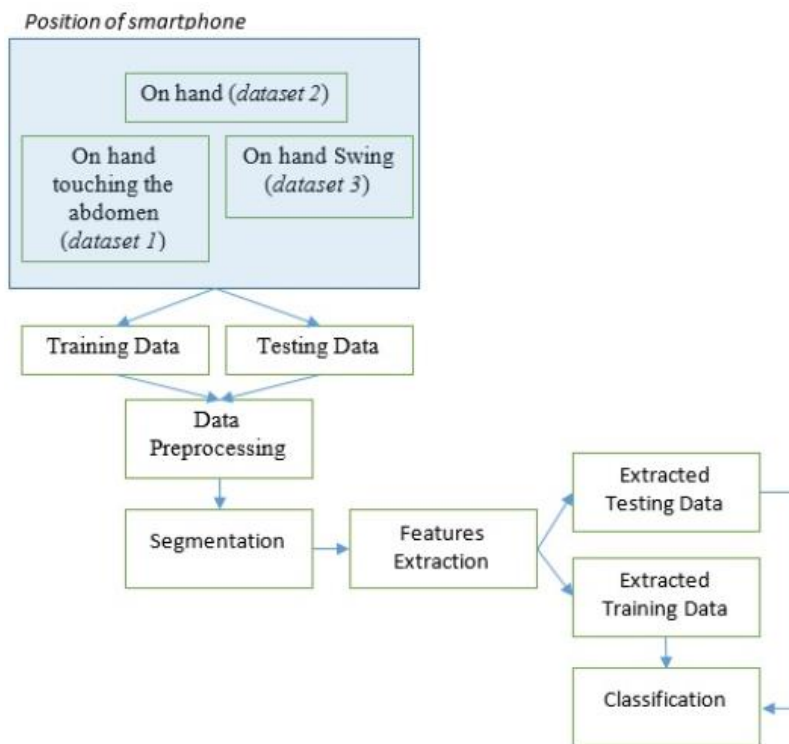


Figure. 7 The framework of the proposed experiment

been conducted in determining the best linear interpolation factor. The result can be seen on Table 2.

The interpolated data is then filtered by centering around zero for all axis. The filtered data is then segmented using Fixed Size Overlapping Sliding Window with window size equivalent to 32.

According to our research in this dataset, window size and linear interpolation rate does play a significant role in producing the result. Having too high of linear interpolation will make the classification accuracy low. It is also same with the window size.

Learning model is generated by OvO concept from 30 different classes for each experiment.

For classifier, the first method used is K-nearest neighbor. The nearest neighbor search algorithm is using linear nearest neighbor search with Euclidean distance as objects distance in the data model. The k value is 1.

The second method is MLP. The number of hidden layer is 20. The learning rate is 0.3 with momentum which is 0.2.

For SVM, the coefficient is set to 0 with the cost is set to 1. The EPS (tolerance of the termination criterion), loss (epsilon for the loss function) and degree of the kernel are set to 0.001, 0.1 and 3 respectively.

For the fourth classifier which is the J48, the parameters are seed which is set to 1 and *numFolds* which is set to 3. Others are left by default.

After training based on the respective classifier algorithm, altogether, there are 435 generated learning models in each classifier.

Accuracy is calculated based on the percentage score of the recall using test dataset. It is produced by calculating the majority score of votes from the 435 generated confusion matrix.

According to Table 1, it can be seen that dataset 1 produced the best accuracy result which yields 96.7% by k-NN with only 1 mistake. This could be due to its stability which it has the fewest shake among other positions. 100% accuracy could not be obtained in this dataset due to one of the subject's data in the testing phase is tempered.

In dataset 2 and dataset 3, the accuracy dropped to 83.3% and 76.7% respectively due to the signals that become more scattered in dataset 2 and the worst in dataset 3 as shown in Fig. 4.

From the Table 2, it can be seen that using single magnitude which has been adopted by [2,9-10], does reduce the overall accuracy. In dataset 1, the accuracy dropped from 96.7 % (3-axis) to 83.3 % (single magnitude). The regression of the accuracy can be seen in all of the datasets. In the overall position, using single magnitude only achieved 43.3% which is very low for accuracy. This could be

Table 1. Overall accuracy on all handheld position using k-NN

| Smartphone position | Dataset 1 | Dataset 2 | Dataset 3 |
|----------------------------------|-------------|-----------|-----------|
| Correct Recognition Rate (CRR) | 29 | 25 | 23 |
| Incorrect Recognition Rate (IRR) | 1 | 5 | 7 |
| Accuracy (%) | 96.7 | 83.3 | 76.7 |

Table 2. Accuracy of classification using k-NN on the single magnitude application on handheld based smartphone sensor placement

| Dataset | 1 | 2 | 3 |
|---|-------------|------|------|
| 3-axis (%) | 96.7 | 83.3 | 76.7 |
| Single Magnitude (%) | 83.3 | 76.7 | 70.0 |
| Single Magnitude (%) on overall dataset | 43.3 | | |

Table 3. Accuracy of classification using k-NN depending on the interpolation size

| Interpolation Size (Hz) (samples/sec) | 25 (No interpolation) | 50 (Factor 1) | 100 (Factor 2) | 200 (Factor 3) | 400 (Factor 4) |
|---------------------------------------|-----------------------|---------------|----------------|----------------|----------------|
| Accuracy (%) | 73.3 | 76.7 | 90.0 | 93.3 | 83.3 |

Table 4. Comparison of accuracy between single classifier setup and OvO multiclass structure using k-NN

| Classification type | Single | | | | OvO | | | |
|---------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 |
| Smartphone position | | | | | | | | |
| CRR | 28 | 24 | 22 | 326 | 29 | 25 | 23 | 343 |
| IRR | 2 | 6 | 8 | 82 | 1 | 5 | 7 | 65 |
| Accuracy (%) | 93.3 | 80 | 73.3 | 80 | 96.7 | 83.3 | 76.7 | 84 |

Table 5. Overall accuracy on all classifier depending on the handheld position

| Classifier | k-NN | | | MLP | | | SVM | | | J48 | | |
|---------------------|-------------|------|------|-------------|-------------|-------------|------|-------------|----|------|-------------|-------------|
| | D1 | D2 | D3 | D1 | D2 | D3 | D1 | D2 | D3 | D1 | D2 | D3 |
| Smartphone Position | | | | | | | | | | | | |
| CRR | 29 | 25 | 23 | 29 | 26 | 25 | 23 | 26 | 21 | 28 | 26 | 25 |
| IRR | 1 | 5 | 7 | 1 | 4 | 5 | 7 | 4 | 9 | 2 | 4 | 5 |
| Accuracy (%) | 96.7 | 83.3 | 76.7 | 96.7 | 86.7 | 83.3 | 76.7 | 86.7 | 70 | 93.3 | 86.7 | 83.3 |

Table 6. Overall accuracy on all classifier for dataset 4 from Ngo [5]

| Classification type | Single | | | | OvO | | | |
|---------------------|--------|------|------|------|------|------|------|------|
| | k-NN | MLP | SVM | J48 | k-NN | MLP | SVM | J48 |
| CRR | 326 | 348 | 320 | 346 | 343 | 362 | 326 | 360 |
| IRR | 82 | 60 | 88 | 62 | 65 | 46 | 82 | 48 |
| Accuracy (%) | 80 | 85.3 | 78.4 | 84.7 | 84 | 88.7 | 79.8 | 88.2 |

due to the loss of the richness in the features before the training and classification. Combining the values of 3 axis into one value reduce the significant features of the dataset. It can be concluded that using this method is not suitable for handheld based sensor gait identification. However, if this method still chosen to be applied, additional methods or treatment need to be done on the dataset in-order to get a better accuracy rate.

The relationship of interpolation with classification accuracy is also analyzed. According to Table 3, the interpolation size plays a significant impact in producing the best accuracy rate. The result of the accuracy was implemented using K-nearest neighbor. It can be seen that higher interpolation produce better result but it regress when more interpolation is added up. It seems that by determining the linear interpolation factor, it does influence the overall classification accuracy as each factor produce different number of datasets and different sets of data distribution. Too much application of linear interpolation may results in overfitting which it would end up in low accuracy [27]. This method is suitable in regenerating new rows of datasets if the captured data is small (captured in small walking distance).

In the comparison between single classifier as from the Table 4, it can be seen that using OvO generally increases the overall accuracy for all types of dataset if compared to single classifier method by [2,6,8,16]. This is due to the successful separation of the overlapped multi class into an easier separable binary class which makes the classes to be more unique and increase the probability of the uniqueness of the dataset. However, it can be seen that using OvO thus increase the learning time as it needs to train many models for each pair of classes.

For the best classifier that works best in OvO arrangement in this experiment, it can be seen that from the Table 5 and Table 6, MLP is the best in all of the positions which produced the highest accuracy score consistently. This is due to the efficiency of the errors and weights adjustment when dealing with continuous or discrete values

[28]. The second best classifier is J48 which scored high in dataset 2 and 3. k-NN only good in dataset 1. In most of the positions, the performance of the classification regress except for SVM which in dataset 2, SVM performs better than dataset 1.

5. Conclusion and Future Works

In this study, gait recognition has been implemented using mobile smartphone which is hand held. Data from three different position has been collected and analyzed. Single magnitude has been investigated in the handheld based smartphone application. Besides that, the factor of the linear interpolation also has been investigated in-order to find the best factor for sampling. The best machine learning also has been identified to works best with OvO classification model for this application.

In future work, the methods of aggregation could be further exploited in finding the best classification result for the OvO classification layout. Besides that, the number of data used could be increased more for a better insights as this is just a small simulation or a stepping stone to a bigger and robust application. Another future work is to optimize the current OvO algorithm especially on the classifier so that the time taken on the training could be reduced.

References

- [1] C. Nickel, "Accelerometer-based biometric gait recognition for authentication on smartphones," Ph.D. dissertation, Technische Universitat, 2012.
- [2] M. Derawi and P. Bours, "Gait and activity recognition using commercial phones," *Comput. Secur.*, Vol. 39, No. 2, pp. 137–144, 2013.
- [3] B. Sun, Y. Wang, and J. Banda, "Gait Characteristic Analysis and Identification Based on the iPhone's Accelerometer and Gyrometer," *Sensors*, Vol. 14, No. 9, pp. 17037–17054, 2014.
- [4] T. Hoang, D. Choi, and T. Nguyen, "Gait authentication on mobile phone using biometric cryptosystem and fuzzy commitment scheme," *Int. J. Inf. Secur.*, Vol. 14, No. 6, pp.

- 549–560, 2015.
- [5] T. T. Ngo, Y. Makihara, H. Nagahara, Y. Mukaigawa, and Y. Yagi, “The largest inertial sensor-based gait database and performance evaluation of gait-based personal authentication,” *Pattern Recognit.*, Vol. 47, No. 1, pp. 228–237, 2014.
- [6] C. Nickel and C. Busch, “Classifying accelerometer data via hidden Markov models to authenticate people by the way they walk,” *IEEE Aerosp. Electron. Syst. Mag.*, Vol. 28, No. 10, pp. 29–35, 2013.
- [7] J. Frank, S. Mannor, J. Pineau, and D. Precup, “Time Series Analysis Using Geometric Template Matching,” *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 35, No. 3, pp. 740–754, 2013.
- [8] Y. Ren, Y. Chen, M. C. Chuah, and J. Yang, “User Verification Leveraging Gait Recognition for Smartphone Enabled Mobile Healthcare Systems,” *IEEE Trans. Mob. Comput.*, Vol. 14, No. 9, pp. 1961–1974, 2015.
- [9] D. Gafurov, E. Snekenes, and T. E. Buvarp, “Robustness of Biometric Gait Authentication Against Impersonation Attack,” *Move to Meaningful Internet Syst. 2006 OTM 2006 Work.*, pp. 479–488, 2006.
- [10] D. Gafurov, “A survey of biometric gait recognition: Approaches, security and challenges,” *Annu. Nor. Comput. Sci. Conf.*, pp. 19–21, 2007.
- [11] S. Sprager and M. B. Juric, *Inertial sensor-based gait recognition: A review*, Vol. 15, No. 9, pp. 22089–22127, 2015.
- [12] T. Hoang, D. Choi, and T. Nguyen, “Gait authentication on mobile phone using biometric cryptosystem and fuzzy commitment scheme,” *Int. J. Inf. Secur.*, Vol. 14, No. 6, pp. 549–560, 2015.
- [13] S. Sprager and D. Zazula, “A cumulant-based method for gait identification using accelerometer data with principal component analysis and support vector machine,” *WSEAS Trans. Signal Process.*, Vol. 5, No. 11, pp. 369–378, 2009.
- [14] M. O. Derawi, C. Nickely, P. Bours, and C. Busch, “Unobtrusive user-authentication on mobile phones using biometric gait recognition,” in *Proceedings - 2010 6th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIHMSP 2010*, 2010, pp. 306–311.
- [15] C. Nickel, C. Busch, S. Rangarajan, and M. Möbius, “Using Hidden Markov Models for Accelerometer-Based Biometric Gait Recognition,” *2011 IEEE 7th Int. Colloq. Signal Process. its Appl.*, pp. 58–63, 2011.
- [16] T. Hoang, T. Nguyen, C. Luong, S. Do, and D. Choi, “Adaptive cross-device gait recognition using a mobile accelerometer,” *J. Inf. Process. Syst.*, Vol. 9, No. 2, pp. 333–348, 2013.
- [17] M. Muaaz and C. Nickel, “Influence of different walking speeds and surfaces on accelerometer-based biometric gait recognition,” *2012 35th Int. Conf. Telecommun. Signal Process. TSP 2012 - Proc.*, pp. 508–512, 2012.
- [18] H. M. Thang, V. Q. Viet, N. Dinh Thuc, and D. Choi, “Gait identification using accelerometer on mobile phone,” *2012 Int. Conf. Control. Autom. Inf. Sci.*, pp. 344–348, 2012.
- [19] S. M. D. P. Jordan Frank, J. Frank, S. Mannor, and D. Precup, “Activity and Gait Recognition with Time-Delay Embeddings Time-Delay Embeddings,” *AAAI Conf. Artif. Intell.*, pp. 407–408, 2010.
- [20] H. Lu, J. Huang, T. Saha, and L. Nachman, “Unobtrusive Gait Verification for Mobile Phones,” *Proc. 2014 ACM Int. Symp. Wearable Comput. - ISWC '14*, pp. 91–98, 2014.
- [21] E. Meijering, “A chronology of interpolation: From ancient astronomy to modern signal and image processing,” *Proc. IEEE*, Vol. 90, No. 3, pp. 319–342, 2002.
- [22] A. Fernández, M. J. Del Jesus, and F. Herrera, “Multi-class imbalanced data-sets with linguistic fuzzy rule based classification systems based on pairwise learning,” in *Computational Intelligence for Knowledge-Based Systems Design*, Springer, 2010, pp. 89–98.
- [23] G. Anthony, H. Gregg, and M. Tshilidzi, “Image classification using SVMs: one-against-one vs one-against-all,” *arXiv Prepr. arXiv0711.2914*, 2007.
- [24] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, “Label ranking by learning pairwise preferences,” *Artif. Intell.*, Vol. 172, No. 16, pp. 1897–1916, 2008.
- [25] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, “An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes,” *Pattern Recognit.*, Vol. 44, No. 8, pp. 1761–1776, 2011.
- [26] J. Friedman, “Another approach to polychotomous classification,” techreport, Department of Statistics, Stanford, CA, 1996.
- [27] M. Bramer, “Using J-pruning to reduce overfitting in classification trees,” *Knowledge-Based Syst.*, Vol. 15, No. 5, pp. 301–308, 2002.

- [28] S. Haykin and N. Network, "A comprehensive foundation," *Neural Networks*, Vol. 2, No. 2004, 2004.