Analysis of Gait Motion Sensor Mobile Authentication with Machine Learning

Sara Kokal[1], Mounika Vanamala[2], Rushit Dave[3]
Computer Science Department, University of Wisconsin-Eau Claire, Eau Claire, U.S.A[1, 2]
Computer Information Science Department, Minnesota State University, Mankato, Mankato, U.S.A[3]

***Abstract*—In recent decades, mobile devices have evolved in potential and prevalence significantly while advancements in security have stagnated. As smartphones now hold unprecedented amounts of sensitive data, there is an increasing need to resolve this gap in security. To address this issue, researchers have experimented with biometric-based authentication methods to improve smartphone security. Following a comprehensive review, it was found that gait-based mobile authentication is under-researched compared to other behavioral biometrics. This study aims to contribute to the knowledge of biometric and gait-based authentication through the analysis of recent gait datasets and their potential with machine learning algorithms. Two recently published gait datasets were used with algorithms such as Random Forest, Decision Tree, and XGBoost to successfully differentiate users based on their respective walking features. Throughout this paper, the datasets, methodology, algorithms, experimental results, and goals for future work will be described.**

***Keywords*—Machine learning; machine learning algorithms; behavioral biometrics; gait dynamics; motion sensors**

I. Introduction

The demand for mobile device performance continues to increase as society and industry becomes more technology oriented. Nowadays, smartphones are used for an ever-expanding array of problems including navigation, calculations, photography, and socialization. The ability to combine solutions to multiple daily functionalities into the applications of a smart device is expected by today's mobile device users. Recently, the use of mobile financial transaction options and the holding of sensitive card data such as Apple Pay, Apple Wallet, PayPal, and Venmo have become popular. In the United States, 59% of in-person stores, restaurants, and other services allow for apple pay, only superseded by 70% in the U.K. [1]. While only needing to bring a phone into a store to complete transactions is appealing to consumers, financial security consequences arise if devices are stolen and broken into. Losing a phone can now have a similar impact to losing a wallet. With these advancements, it has been necessary to find secure ways to protect the sensitive data smart devices hold.

In response to these concerns, researchers have been investigating the potential of novel authentication methods to improve mobile device security. The two current most common methods of authentication for devices are knowledge-based and physiological-based. In knowledge-based authentication, information that is known only to the owner is used to secure the device. This method can be deployed as a sequence of characters and numbers, or as a graphical pattern. While knowledge-based authentication is widely popular and easy to use, it is also prone to security risks if this information is leaked or stolen by an adversary [2]. Physiological biometrics uses physical traits of the user for authentication, such as facial, fingerprint, palm or ocular characteristics scanned by the device. These methods have become more popular in recent years and have become implemented in phones and other devices. Unfortunately, physiological methods have found to be less accurate and more costly than expected, sometimes requiring additional hardware to accurately scan the user's features [3]. Researchers have found an alternative solution in the form of behavioral biometrics. Behavioral biometrics uses an individual's unique behavioral characteristics to secure a device. They are cost effective, as they collect data with low-cost sensors already within the device such as motion sensors and the touch screen [2]. It is also notable that while knowledge-based and physiological methods are generally used as a one-time authentication strategy, behavioral biometrics methods can continuously authenticate the device while it is being used. This strategy analyzes user behavior repeatedly to secure the device in the case that an initial one-time authentication has failed, and the device has already been accessed [4].

There are many different behavioral strategies used to secure a device with the innate sensors, including touch dynamics, keystroke dynamics, and motion dynamics. Motion dynamics utilize the motion sensors in a device, including the accelerometer, gyroscope, and magnetometer sensors. Motion dynamics can be captured anytime the device is being used where motion is involved. One subset is known as gait dynamics, where the device records data from the motion sensors while the user walks to capture their gait characteristics. As of late, these behavioral biometrics methods have been found to be effective in securing mobile devices when used with machine learning and deep learning algorithms with high accuracy metrics and low error rates [5].

This paper aims to further research into this field of study with these contributions:

- Expand knowledge into behavioral biometrics authentication with the comparison of two recently published gait datasets [6, 7].
- Develop Machine Learning models (Random Forest, Decision Tree, XGBoost) to evaluate the efficiency of gait biometric authentication and compare classifier results.

II. Background and Related Work

The direction of this study was inspired by the findings of a past work, reviewing the use of Machine Learning (ML) and Deep Learning (DL) algorithms with biometrics-based mobile authentication systems [5]. This review examined 66 of the latest experimental studies on behavioral biometrics with touch dynamics, keystroke dynamics, motion dynamics and gait dynamics with a focus on how they performed with various algorithms. It was found that studies on the usage of AI algorithms with biometrics have become popular in recent years as the increase in number and quality of public training datasets has allowed for the construction of better performing and more accurate models. Of the dynamics listed, touch dynamics and motion dynamics were the most popular, with 24 and 18 studies cited respectively. Despite having decent performance metrics in comparison, gait dynamics were found to be under-researched, numbering at 11 cited studies, the lowest of the four dynamics. Therefore, this study has sought to breach this gap by analyzing the performance of recently published gait datasets with AI algorithms.

In previous reviews [5, 8], it was established that to continue progress in the investigation of behavioral biometrics mobile authentication systems, it is worthwhile to focus on how systems can be advanced past previous boundaries and ensure models can hold up against real world contexts. One way to do this is to ensure datasets have larger sample sizes that can properly represent a population and effectively train a ML/DL model. In recent years, many high-quality biometrics datasets with larger sample sizes have been published for public use, allowing us to advance model quality. One example in gait dynamics would be the IDNet dataset, published in 2018 [9]. This dataset has since been cited in over 200 papers with a majority published after the year 2020. The IDNet dataset consists of accelerometer and gyroscope data collected from 50 subjects over a six-month period and was collected to classify gait cycles regardless of device orientation. Of the reviewed studies, [10], [11], and [12] used the IDNet dataset to evaluate various LSTM-based models and resulted in accuracy metrics ranging from 96-99%. Another notable dataset would be the WhuGait dataset, published in 2020 [13]. This dataset contained gait motion sensor data from 118 individuals collected in an unrestrained "wild" environment. Their presenting study analyzed the dataset performance with a hybrid Convolutional Neural Network (CNN) and Long-Short Term Memory (LSTM) model, resulting in an accuracy performance of 93.75%.

Mobile gait authentication studies typically rely on the use of motion sensors within the phone such as accelerometer, gyroscope, and magnetometer to capture an individual's gait cycle characteristics. Of the motion dynamics studies reviewed, pairing accelerometer and gyroscope sensor data was the most popular [5]. Within gait studies, a similar pattern was seen with studies preferring either accelerometer data alone or a pairing of accelerometer and gyroscope data. In the WhuGait study [13], accelerometer and gyroscope data were collected. Results from this study found that individually, accelerometer data performed better than gyroscope, but using both was complementary.

Overall, recent studies in mobile gait authentication favored hybrid Deep Learning (DL) models. Of the gait studies reviewed, architectures using CNN feature extraction with LSTM classification numbered half of the cited papers with accuracy metrics ranging from 90.00-99.99% [5]. Within some of these studies, the hybrid models were also compared to ML algorithms in performance. In all the studies, DL algorithms outperformed ML algorithms, but in some the ML algorithms performed at adequate levels comparatively. In the IDNet paper, a model with CNN feature extraction and One-Class Support Vector Machine (OC-SVM) classification was tested on their data with a performance of < 0.15 False Acceptance Rate (FAR) and False Rejection Rate (FRR) [9]. In another study [14], a CNN model was proposed for gait authentication and evaluated with a large public dataset. Their model was compared with the performance of Random Forest (RF) and K-Nearest Neighbors (KNN) algorithms. CNN had the best performance with 0.9882 accuracy, but RF did not lag too far behind with an accuracy of 0.9551. In a third study [15], walking data from a small dataset was tested on LSTM, CNN, Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) in two scenarios. In the binary classification scenario using training data from both the target user and other users, SVM had the best performance compared to MLP with 98.42% accuracy. In a scenario where the training data only included the target user's data, LSTM significantly outperformed SVM with 90.24% accuracy. Overall, DL algorithms have proved to dominate current gait mobile authentication studies with high accuracy rates and low error rates, but it has been noted in some comparison studies that ML algorithms such as RF and SVM remain effective in certain scenarios. This can prove useful if one is attempting to build a smaller security system with less data than is required for advanced DL models.

Studies most recently published demonstrating the continued relevance of gait dynamics mobile authentication research include [16], and [17]. In study [16], researchers collected accelerometer and angular velocity sensor readings from 10 individuals in pocket and hand-hold positions over periods of around 90 seconds. They trained a CNN model with the data, producing an average accuracy of 0.9175. The study concluded that gait data collected over short periods of time can be successfully used for authentication. In study [17], researchers proposed IRGA, their implicit real-time gait authentication system using a hybrid CNN+LSTM model. They collected accelerometer, gyroscope, and magnetometer sensor readings from 16 individuals in varying positions and walking styles, analyzing the impact of constrained vs unconstrained environments. They concluded that authentication based on gait characteristics is feasible despite limitations. Their model was tested on multiple datasets, achieving a highest average accuracy of 99.4% with the ZJU-GaitAcc dataset.

III. Methodology

A. Datasets

Two datasets were chosen to compare algorithm performance. They were each chosen for their similarities as well as their relatively recent publishing dates bearing a limited number of citations. The first, BB-MAS, is a large dataset

comprising of swiping, keystroke, and gait data collected from desktop, tablet, and mobile phone devices [2]. It was published in 2019 by Belman et al. The dataset demographic consists of 117 individuals, 72 male and 45 female, of which the majority spoke English and was right hand dominant. The data collection process consisted of a sequence of events each individual performed to complete all dynamics activities. First, the individual would start the desktop and touch dynamics activities before walking down a corridor with their mobile device, passing through a stairwell, walking down another corridor, and returning along the same path. The files were split between device used and sensor collected from as well as device position. Gait accelerometer and gyroscope data was collected from a mobile device at a 50Hz sampling rate in two positions; one where the device was held in the hand, and one where the device was placed in the pocket. The X, Y, and Z axis values were recorded for each sensor. Gait data collection time for each individual ranged around 5-10 minutes. The mobile devices used in the study were Samsung-S6 and HTC-One phones. Timestamps were included along with each user file folder to differentiate between corridor walking and stair climbing. Only data in which the individual was performing walking movements along a corridor with a mobile device was used.

The second dataset, MMUISD, was published in 2020 by Permatasari et al [3]. The MMUISD dataset originally consisted of data from 322 undergraduate students (246 male and 76 female) which was cut down to 120 for the publicly available dataset. The data collection process was simple, requiring individuals to walk down a 15-meter corridor with their device. An android application was downloaded onto each device and used to collect accelerometer and gyroscope data at a 50Hz fixed sampling rate. X, Y and Z axis values were recorded for both accelerometer and gyroscope. There were six different device positions in the study, of which only the hand and pocket positions were used. Users were instructed to walk naturally without restraints in three different speeds: slow, normal, and fast. User file data was differentiated based on speed and position. Data collection time ranged from 5-8 minutes for each individual to complete all speeds. Due to time constraints, the number of individual users per speed and position in the public dataset differed between 65 and 99 individuals as can be seen in Table I.

TABLE I. MMUISD PARTICIPANTS

Position / Speed	# of Participants
Left H slow	65
Left H Normal	99
Left H Fast	96
Right H Slow	79
Right H Normal	80
Right H Fast	76
Left P Slow	90
Left P Normal	74
Left P Fast	97
Right P Slow	96
Right P Normal	75
Right P Fast	75

B. Data Cleaning and Preprocessing

Before feature extraction, it is important to properly preprocess and clean the data to prevent avoidable errors. The pandas python library and PyCharm environment were used to facilitate these steps. Both datasets selected had clear signals without significant outliers, so it was not needed to take many steps in the initial cleaning process. The null values in all rows were replaced with 0 for all user files in each dataset.

The preprocessing steps were unique to each dataset since the organization of the user files and data signals differed slightly. The MMUISD dataset was straightforward, as both the gyroscope and accelerometer sensor readings were compiled in the same file for each user and only recorded walking data. BB-MAS instead separated gyroscope and accelerometer readings into different files. Due to how the data was collected, stair climbing and walking were recorded on the same files and required given timestamps to differentiate the two. Taking extra steps to preprocess the BB-MAS files was necessary to properly compare both datasets. First, the timestamp file matching the current user file being preprocessed was extracted and the checkpoints corresponding to the walking segments were identified. Then, the accelerometer and gyroscope signal files were merged based on the recording times. Using the checkpoints, walking sequence data was separated and concatenated into a new Data Frame to be used in the feature extraction process.

C. Feature Extraction

In time series analysis problems, time domain features are typically extracted from sequences of the recorded data. The sequence lengths were chosen by visualizing the mean of

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

From the x, y, and z axis of each signal with respect to the time. An example of this visualization is provided by Fig. 1. For the MMUISD dataset, a sequence length of 10 was chosen. The BB-MAS dataset has a greater amount of datapoints, thus a sequence length of 20 was found to be optimal.

The same feature sets were chosen for both datasets for comparison purposes. Eight different statistical features were extracted from the x, y, z axes and m of both the accelerometer and gyroscope signal. In total, 64 features were extracted from each user file. The features were selected based on previous studies as well as the recommendations of the chosen datasets. In the BB-MAS readme document, Mean, Standard deviation, Band Power, Energy, Median Frequency, Interquartile Range, Range, Signal to Noise Ratio, Dynamic Time Warping Distance, Mutual Information and Correlation were suggested as possible gait features. Other gait authentication studies reviewed commonly included features such as Mean, Standard Deviation, Band Power, Median Frequency, Interquartile Range, Range, Dynamic Time Warping Distance, Average Max and Min, Root Mean Square, and Average Absolute Difference [9, 18, 19]. For the final feature set, the Mean, Standard Deviation, Average Min and Max, Interquartile Range, Range, Root Mean Square, and Absolute Deviation from x, y, z and m were extracted.

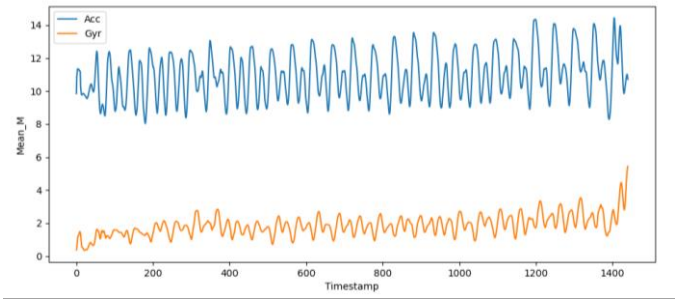

Fig. 1. Visualization of the mean m of the accelerometer and gyroscope signals from a user in the MMUISD dataset.

D. Training and Testing

Data from each user was split with 80% used for training the models and 20% used for testing. This 80/20 split was chosen as 80/20 and 70/30 splits for training and testing sets have been found by empirical research studies to be optimal for statistical model performance [20]. The authentic user data and imposter data was then concatenated together for the final training and testing sets. For the testing set, the user data was concatenated with a 40% random sample of the imposter data to prevent overfitting and bias. To enable the model to properly differentiate an authentic user from an imposter within the training data, each data point included a class label with a 0 or 1. A 0 represented an authentic user and while a 1 was an imposter. During the testing process, these labels were used to determine how accurate the classifier's decision making was. For data normalization, Standard Scaler was used for the Random Forest and Decision Tree Models, while Simple Imputer was used with the XGBoost model.

During the initial testing process, the parameters of each ML model were fine tuned to produce the best classification results with the datasets. The Random Forest model comprised of a parameter set with 100 estimators, a max depth of 20, a minimum sample split of 2, a minimum number of trees of 1, 7 jobs to run in parallel, and class weights determined by the number of positive and negative samples. The Decision Tree model included Gini Impurity function, a max depth of 10, and class weights determined by the number of positive and negative samples. The XGBoost model required more manipulation than the previous models, producing higher levels of overfitting. To combat this, the feature set was cut down to around 15 by evaluating feature importance with a basic binary logistic XGBoost model. Feature importance was visualized with a pyplot bar graph, and features that produced an importance level of less than 0.2 were removed. Features that produced high levels of feature importance in both datasets included Min and Max, Mean, Root Mean Square, and Range. The final XGBoost model parameters included binary logistic objective, a learning rate of 1.5 and a scale pos weight determined by class balance.

IV. Results

This study intends to evaluate the efficiency of gait characteristics for differentiating mobile users by comparing the classification performance of ML algorithms with two recent gait datasets. For classification analysis, high performance binary classifiers were selected such as Random Forest, Decision Tree, and XGBoost. The classifiers were trained and evaluated as specified in the previous section on all users.

To properly evaluate the performance of the models on the datasets, the following statistical evaluation metrics were included in the results for each user in each dataset:

- The Accuracy (ACC): Rate of correctly predicted results.
- F-Score (F1): Measure of the harmonic mean of precision and recall.
- False Positive Rate (FPR): Rate of incorrectly identified authentic users.
- False Negative Rate (FNR): Rate of incorrectly identified imposters.
- Equal Error Rate (ERR): Threshold where FPR and FNR are equal.

When observing these metrics, lower EER, FPR and FNR rates are desired over higher ones, as they represent how well a model can differentiate between authentic users and imposters. The accuracy metric is helpful for measuring overall model performance accuracy. Similarly, a larger F1 score is indicative of strong overall model performance.

Table II shows the results from training the models with the MMUISD dataset. Random Forest had the best overall classification performance using MMUISD with an average accuracy of 98.90% and an average EER of 4.18%. Random Forest achieved the highest accuracy in the right pocket position at slow speed with 99.18% and a lowest EER of 2.76% in the right pocket at fast speed. While the XGBoost model achieved a higher average accuracy than Random Forest with 98.98%, it also had higher average error rates of 18.94%. DT had a lowest error rate with 3.69% but had a smaller overall average accuracy than Random Forest. The XGBoost model had tended to overfit to the user, resulting in higher and more varied error rates after tuning.

BB-MAS results are shown in Table III. Random Forest had the best performance with an overall accuracy of 99.03% and an EER of 1.04%. Decision Tree and XGBoost had similar differences in performances with Decision Tree achieving lower accuracy scores but a similar EER score. XGBoost again achieved the highest accuracy score but with higher EER scores due to a tendency to overfit the data.

Table IV compares the performance of the two datasets. In both datasets, Random Forest had the best overall performance. The Pocket Phone position achieved the best accuracy and EER results in both datasets as well. One noticeable difference is that Decision Tree achieved a better accuracy in the hand position with the MMUISD dataset, with a score of 95.41% compared to 88.98% with the BB-MAS dataset. It is also notable that the pocket position achieved better error results with the BB-MAS dataset compared to the MMUISD dataset, with an average EER of 1.04% compared to 3.37% when evaluated with the Random Forest model.

TABLE II. MMUISD RESULTS

DT						RF						XGB					
	ACC	*F1*	*FPR*	*FNR*	*EER*		*ACC*	*F1*	*FPR*	*FNR*	*EER*		*ACC*	*F1*	*FPR*	*FNR*	*EER*
LHF	0.955748	0.976076	0.033123	0.044878	0.033123	**LHF**	0.98746608	0.99340466	0.03972127	0.01166933	0.03972127	**LHF**	0.98922657	0.99435532	0.24901326	0.00476506	0.2385966
LHN	0.954952	0.975971	0.034801	0.045433	0.034801	**LHN**	0.98684145	0.99316609	0.04872764	0.01234906	0.04872764	**LHN**	0.98936387	0.9941629	0.22193526	0.00541354	0.21183425
LHS	0.947999	0.972288	0.051925	0.052046	0.051925	**LHS**	0.98612967	0.99281256	0.07339557	0.01245069	0.07339557	**LHS**	0.98534627	0.99179774	0.19992959	0.00746016	0.19992959
RHF	0.957313 39	0.976906 06	0.029515 84	0.043327 65	0.029515 84	**RHF**	0.98682726	0.99310342	0.04148482	0.01229703	0.04148482	**RHF**	0.99153528	0.99562666	0.22648091	0.00181062	0.22648091
RHN	0.963692	0.980578 18	0.031533 69	0.036576 86	0.031533 69	**RHN**	0.98808698	0.99377241	0.03478625	0.01123549	0.03478625	**RHN**	0.98263 13	0.98957693	0.17738651	0.01226052	0.16488651
RHS	0.945241 12	0.970395 49	0.044671 54	0.055098 45	0.044671 54	**RHS**	0.98671675	0.99306046	0.06145487	0.01171924	0.06145487	**RHS**	0.99139401	0.99555724	0.22672113	0.00209414	0.2140629
LPF	0.956780 03	0.976899 44	0.028857 21	0.043725 51	0.028857 21	**LPF**	0.99082847	0.99523198	0.02823727	0.00868765	0.02823727	**LPF**	0.99152836	0.99558237	0.21429292	0.00323139	0.19367436
LPN	0.953753 27	0.975063 13	0.035279 54	0.046919 68	0.035279 54	**LPN**	0.99180405	0.99570664	0.0283168	0.00760468	0.0283168	**LPN**	0.99283632	0.9962931	0.17612441	0.00181231	0.14909739
LPS	0.956781 19	0.976991 4	0.038229 65	0.043446 12	0.038229 65	**LPS**	0.99155894	0.99562532	0.04358728	0.00759491	0.04358728	**LPS**	0.99062831	0.99495116	0.18969212	0.00474206	0.17858101
RPF	0.946338 84	0.971154 74	0.031079 17	0.054377 61	0.031079 17	**RPF**	0.99031528	0.9949851	0.02765891	0.00920545	0.02765891	**RPF**	0.9926335	0.99619316	0.19892363	0.00164841	0.15892363
RPN	0.945390 75	0.970535 21	0.037292 41	0.055319 01	0.037292 41	**RPN**	0.99026087	0.9949103	0.03528433	0.00892481	0.03528433	**RPN**	0.98612486	0.9916437	0.18623141	0.008704	0.17289808
RPS	0.950229 56	0.973125 76	0.041392 41	0.050157 63	0.041392 41	**RPS**	0.99185715	0.99576096	0.03915885	0.00712121	0.03915885	**RPS**	0.99440606	0.99712603	0.16481417	0.00154831	0.16481417
Hand Avg	0.954157 59	0.975369 12	0.037595 01	0.046226 66	0.037595 01	**Hand Avg**	0.98701136	0.99321993	0.0499284	0.01195347	0.0499284	**Hand Avg**	0.98824955	0.9935128	0.21691111	0.00563401	0.20929846
Pocket Avg	0.951545 61	0.973961 61	0.035355 07	0.048990 93	0.035355 07	**Pocket Avg**	0.99110413	0.99537005	0.03370724	0.00818978	0.03370724	**Pocket Avg**	0.99135957	0.99529825	0.18834644	0.00361441	0.16966477
Right Avg	0.951367 61	0.973782 57	0.035914 18	0.049142 87	0.035914 18	**Right Avg**	0.98901072	0.99426544	0.03997134	0.01008387	0.03997134	**Right Avg**	0.98902638	0.99387832	0.16588691	0.00637815	0.15491468
Left Avg	0.954335 58	0.975548 16	0.037035 9	0.047367 52	0.037035 9	**Left Avg**	0.98910478	0.99432454	0.04366431	0.00999757	0.04366431	**Left Avg**	0.98982162	0.99452377	0.20849793	0.00457075	0.19528553
Final Avg	0.952351 92	0.974393 74	0.036977 65	0.048099 45	0.036977 65	**Final Avg**	0.98905775	0.99429499	0.04181782	0.01007163	0.04181782	**Final Avg**	0.98980456	0.99440553	0.20262878	0.00462421	0.18948162

TABLE III. BB-MAS RESULTS

	ACC	F1	FPR	FNR	EER
HP DT	0.88989423	0.93963771	0.04713927	0.11153753	0.04713927
PP DT	0.95189307	0.97445261	0.01750336	0.04881871	0.01750336
HP RF	0.98119212	0.99026581	0.04277626	0.01829441	0.04277626
PP RF	0.99037403	0.99503812	0.01046715	0.00962724	0.01046715
HP XGB	0.9941452	0.99701005	0.19952382	0.00152353	0.19952382
PP XGB	0.99547658	0.99766044	0.12219224	0.00198634	0.12219224

TABLE IV. BB-MAS VS MMUISD

	BB-MAS					MMUISD				
	ACC	F1	FPR	FNR	EER	ACC	F1	FPR	FNR	EER
HP DT	0.88989423	0.93963771	0.04713927	0.11153753	0.04713927	0.95415759	0.97536912	0.03759501	0.04622666	0.03759501
PP DT	0.95189307	0.97445261	0.01750336	0.04881871	0.01750336	0.95154561	0.97396161	0.03535507	0.04899093	0.03535507
HP RF	0.98119212	0.99026581	0.04277626	0.01829441	0.04277626	0.98701136	0.99321993	0.0499284	0.01195347	0.0499284
PP RF	0.**99037403**	0.99503812	0.01046715	0.00962724	0.**01046715**	0.**99110413**	0.99537005	0.03370724	0.00818978	0.**03370724**
HP XGB	0.9941452	0.99701005	0.19952382	0.00152353	0.19952382	0.98824955	0.9935128	0.21691111	0.00563401	0.20929846
PPXGB	0.99547658	0.99766044	0.12219224	0.00198634	0.12219224	0.99135957	0.99529825	0.18834644	0.00361441	0.16966477

V. DISCUSSION AND ANALYSIS

The three chosen algorithms had very similar classification performance between the two datasets with slight differences in EER regarding device positioning. The performance of the models did not differ between the between positions and speeds with relation to the number of participants that collected in each position as described in Table I. Between both datasets, Random Forest was found to be the best performing algorithm overall with high accuracy rates paired with lower EER rates.

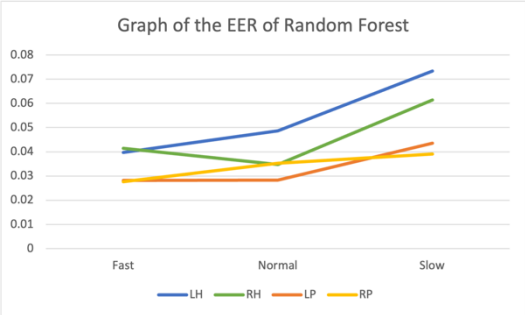

Fig. 2. Visualization difference in EER performance of Random Forest between different phone positions.

The results for the MMUISD dataset have interesting implications on the effect of position and speed on model performance. As shown in Fig. 2, Random Forest EER generally increased as the user's speed became slower. This could imply more variation in the data as the walk speed decreases, as opposed to a fast speed with a lower EER rate. A similar pattern was found in the Decision Tree model. The XGBoost did not follow this pattern as it had more varied error rates. The pocket position achieved generally better results overall compared to the hand position with all algorithms. This could imply that keeping the device closer to the body results in more stability and less variation and noise in the signal compared to holding the device in the hand. The difference in performance was seen more prominently with Random Forest and XGBoost compared to Decision Tree. For Random Forest, the EER was 3.37 % in the pocket compared to 4.99% in the hand. Model accuracy and F1 score did not differ significantly between the right and left positions but EER increased slightly with the left-hand position.

The BB-MAS dataset had nearly identical results to MMUISD as shown in Table IV. Once again, Pocket Phone position achieved better accuracy scores and EER values with all algorithms compared to the hand phone position, emphasizing the possibility that having the device closer to the body provides a more stable and predictable signal for the models. Compared to the MMUISD dataset, with BB-MAS the pocket position had better error results with an average of 1.04% EER with Random Forest.

In Table V, results with the MMUISD and BB-MAS datasets have been compared with recent reviewed studies utilizing datasets of similar participant size [10, 11, 12, 13, 14]. The comparative studies utilized well-known public datasets such as IDNet and WhuGait with various LSTM models. It was observed that the produced results outperformed in average accuracy rates with Random Forest. Most notably, Random Forest trained on MMUISD dataset achieved one of the highest accuracies overall of 0.9911 on similar and higher levels than comparative studies using high performance DL models such as LSTM and CNN. The accuracies with RF were also achieved with suitably low error rates. This is indicative that ML models still have the potential to meet and even exceed the authentication performance of DL models with careful selection of parameters and quality datasets.

TABLE V. COMPARATIVE ANALYSIS

Dataset	MMUISD		BB-MAS		IDNet			WhuGait		Kaggle	
Model	RF	DT	RF	DT	ContAuth LSTM [10]	CNN+LSTM [11]	HDLN LSTM [12]	HDLN LSTM [12]	CNN+LSTM [13]	CNN [14]	RF [14]
Accuracy	**0.99110413**	0.95415759	0.99037403	95189307	0.97	0.977	0.9965	0.9789	0.9375	0.9882	0.9551

VI. LIMITATIONS AND FUTURE WORK

While gait authentication demonstrates potential as a form of behavioral biometrics authentication for mobile devices, it faces limitations that prevent it from logically being used as a sole security method. Gait authentication has a downside in that it requires an individual to move to collect samples. It also faces various obstacles in behavioral variation related to the surrounding environment, such as stairs, hills, and user health [4]. Thus, it is recommended that current gait dynamics authentication methods are used in low security applications as a supporting security method in a multimodal system [4].

One limitation acknowledged in this study would be that the model training strategy utilized is a simplified version that uses only time-domain features extracted directly from the accelerometer and gyroscope sensors and segmented with fixed time intervals. Nowadays, many gait studies are using more advanced methods of characterizing an individual's gait walking pattern [2]. For example, in study [19], the signal was segmented according to the gait cycle instead of a fixed time interval. This was done by using an autocorrelation algorithm to detect the points in the signal at which a heel touch can be identified with the Z-axis signal magnitude. Then, the signal was segmented based on these points. In study [20], a similar strategy was used in which gait cycle segmentation was performed by identifying accelerometer signal change points with autocorrelation coefficients and segmenting based on the identified patterns. From there, a feature vector was extracted from each pattern in time and frequency domains. Due to complexity and time constraints, this study did not utilize these strategies. In the future, it could be beneficial to the expansion of research in gait dynamics authentication if the code for some of these strategies was documented and made accessible for public use and analysis.

Another limitation in this study would be the construction of the XGBoost model. Despite attempts at parameter manipulation and feature analysis, the XGBoost model remained somewhat overfit, resulting in high accuracy at the expense of suitable EER rates. For future research, the XGBoost is not recommended for use with these datasets unless further steps are taken to properly avoid overfitting.

For today's ML and DL models, it is considered best practice to produce a model that can properly represent a diverse population. The datasets chosen for this study, while including a greater number of individuals than used in previous datasets historically, still included bias towards certain groups. For example, both datasets included a larger number of male participants than female. While this study did not test for how

gender bias affected model performance on different individuals, this could be analyzed in future work. Another possible form of bias could be the balance of right-handed and left-handed individuals. In the demographic file of the BB-MAS dataset it can be found that nearly all participants are listed as right-handed.

As established in the background section of this paper, while ML models have been found to perform well with gait dynamics authentication, DL models generally outperform by great margins in both accuracy and error rates. With the results of this study, it was found that ML models such as Random Forest can still match and exceed the performance accuracy of recent studies using DL algorithms while maintaining acceptable error rates. For future work, the next direction of study would be to analyze and compare the performance of DL models such as CNN or LSTM with ML models, using the selected or similar gait datasets. Current trends in research have expanded from ML into the potentials of DL, thus it is encouraged that gait authentication should be further investigated with DL algorithms to advance potential for security. As devices have progressively become mobile in nature, it is necessary to take advantage of motion-sensing in security applications and pursue study in their advancements with both ML and DL algorithms.

VII. Conclusion

From the results of this study, it can be concluded that both datasets perform well with machine learning algorithms to classify gait walking characteristics. The MMUISD dataset may be preferable in a study that aims to observe the effects of different speeds or positions on gait classification performance. The BB-MAS dataset could also be preferable in a study that aims to identify a broader context for behavioral biometrics security including movement and touch interactions across different devices and environments.

After analyzing classifier performance, Random Forest was recognized as an optimal ML classifier for gait dynamics classification capable of achieving similar results to DL models. While XGBoost achieved the highest average accuracy and Decision Tree achieved the lowest average EER rates between datasets, Random Forest resulted in the best overall metrics balancing both categories. In the pocket position, Random Forest had an average accuracy of 99.03% with the BB-MAS dataset and 99.11% with the MMUISD dataset. Random Forest also achieved optimal EER rates below 5% with 1.04% in the pocket position. XGBoost could possibly be manipulated further to combat overfitting and achieve lower error rates.

Through compared analysis of the performance in different scenarios, it has been observed that position and speed can influence classifier performance. In both datasets and all algorithms, placing the device in the pocket position had better accuracy and EER scores compared to the hand position. This could imply that keeping the mobile device in a position closer and secured to the body results in motion signals with more stability and less variation. It was also observed that as the walking speed increased, EER rates increased as well. This could suggest that slower walking speeds can result in more variation in the gait cycle signal, resulting in less favorable algorithm performance. While noticeable, these differences did not differ too significantly, demonstrating the potential for gait dynamics authentication in real world scenarios.

Regardless of these results, in the real world, an individual will not be confined to a set walking speed or corridor. It is recommended that future studies endeavor to build datasets with more variation in position and activity to allow for the construction of feasible gait authentication models in real world contexts. It is hoped that this study can provide worthwhile information to contribute to the advancement of behavioral biometrics mobile authentication models.

Acknowledgment

The funding for this project has been provided by the University of Wisconsin-Eau Claire's Blugold Fellowship and the University of Wisconsin-Eau Claire Computer Science Department's Karlgaard Scholarship.

References

[1] B. Reynor, "Apple Pay usage either for online payments or at POS in various countries worldwide as of November 2023," 2023, Retreived from https://www.statista.com/statistics/1264671/global-apple-pay-adoption/.

[2] C. Wang, Y. Wang, Y. Chen, H. Liu, and J. Liu, "User authentication on mobile devices: Approaches, threats and trends," Computer Networks, vol. 170, pp. 107118, April 2020.

[3] S. Nyle, L. Pryor, and R. Dave, "User authentication schemes using machine learning methods—a review," Springer Singapore, In Proceedings of International Conference on Communication and Computational Technologies: ICCCT 2021, pp. 703-723, 2021.

[4] D. Gabriel, L. Jesus, and M. P. Segundo, "Continuous authentication using biometrics: An advanced review," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 10(4), pp. e1365, 2020.

[5] K. Sara, M. Vanamala, and R. Dave, "Deep Learning and Machine Learning, Better Together Than Apart: A Review on Biometrics Mobile Authentication," Journal of Cybersecurity and Privacy, vol. 3, pp. 227-258, 2021.

[6] A. K. Belman, L. Wang, S. S. Iyengar, P. Sniatala, R. Wright, R. Dora, J. Baldwin, Z. Jin, and V. V. Phoha, "Insights from BB-MAS--A Large Dataset for Typing, Gait and Swipes of the Same Person on Desktop, Tablet and Phone," *arXiv preprint arXiv:1912.02736*, 2019.

[7] P. Jessica, T. Connie, and O. T. Song, "The MMUISD gait database and performance evaluation compared to public inertial sensor gait databases," Springer Singapore, Computational Science and Technology: 6th ICCST 2019, vol. 603, pp. 189-198, August 2019.

[8] K. Sara, L. Pryor, and R. Dave, "Exploration of Machine Learning Classification Models Used for Behavioral Biometrics Authentication," In Proceedings of the 2022 8th International Conference on Computer Technology Applications, pp. 176-182, May 2022.

[9] G. Matteo, and M. Rossi, "Idnet: Smartphone-based gait recognition with convolutional neural networks," Pattern Recognition, vol. 74, pp. 25-37, February 2018.

[10] J. Chauhan, Y. D. Kwon, P. Hui, C. Mascolo, "Contauth: Continual learning framework for behavioral-based user authentication," Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 4, pp. 1-23, 2020.

[11] X. Zeng, X. Zhang, S. Yang, Z. Shi, C. Chi, "Gait-based implicit authentication using edge computing and deep learning for mobile devices," Sensors, vol. 21, pp. 4592, 2021.

[12] Q. Cao, F. Xu, H. Li, "User Authentication by Gait Data from Smartphone Sensors Using Hybrid Deep Learning Network," Mathematics, vol. 10, pp. 2283, 2022.

[13] Z. Qin, Y. Wang, Q. Wang, Y. Zhao, and Q. Li, "Deep learning-based gait recognition using smartphones in the wild," IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3197-3212, 2020.

[14] M. A. Iqbal, S. Roy, S. Mandal, and R. Talukdar, "Privacy protected user identification using deep learning for smartphone-based participatory sensing applications," Neural Computing and Applications, vol. 33, pp. 17303-17313, 2021.

[15] H. Guangyuan, Z. He, and R. B. Lee, "Smartphone impostor detection with behavioral data privacy and minimalist hardware support," *arXiv preprint arXiv:2103.06453*, 2021.

[16] H. Thang, and D. Choi, "Secure and privacy enhanced gait authentication on smart phone," The Scientific World Journal, 2014.

[17] H. Thang, D. Choi, V. Vo, A. Nguyen, and T. Nguyen, "A lightweight gait authentication on mobile phone regardless of installation error," Security and Privacy Protection in Information Processing Systems: 28th IFIP TC 11 International Conference, pp. 83-101, July 2013.

[18] D. Zach, N. Siddiqui, T. Reither, R. Dave, B. Pelto, M. Vanamala, and N. Seliya, "Continuous User Authentication Using Machine Learning and Multi-Finger Mobile Touch Dynamics with a Novel Dataset," IEEE, 2022 9th International Conference on Soft Computing & Machine Intelligence (ISCMI), pp. 42-46, 2022.

[19] J. Choi, S. Choi, and T. Kang, "Smartphone Authentication System Using Personal Gaits and a Deep Learning Model," Sensors, vol. 23, pp. 6395, 2023.

[20] L. Yang, X. Li, Z. Ma, L. Li, N. Xiong, and J. Ma, "IRGA: An Intelligent Implicit Real-time Gait Authentication System in Heterogeneous Complex Scenarios," ACM Transactions on Internet Technology, vol. 23, pp.1-29, 2023.