A. Brömme, C. Busch, A. Dantcheva, C. Rathgeb and A. Uhl (Eds.): BIOSIG 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019

Gait verification using deep learning with a pairwise loss

Vijaya Krishna Yalavarthi[1], Josif Grabocka[2], Hareesh Mandalapu[3], Lars Schmidt-Thieme[4]

Abstract: A unique walking pattern to every individual makes gait a promising biometric. Gait is becoming an increasingly important biometric because it can be captured non-intrusively through accelerometers positioned at various locations on the human body. The advent of wearable sensors technology helps in collecting the gait data seamlessly at a low cost. Thus gait biometrics using accelerometers play significant role in security-related applications like identity verification and recognition. In this work, we deal with the problem of identity verification using gait. As the data received through the sensors is indexed in time order, we consider identity verification through gait data as the time series binary classification problem. We present deep learning model with a pairwise loss function for the classification. We conducted experiments using two datasets: publicly available ZJU dataset of more than 150 subjects and our self collected dataset with 15 subjects. With our model, we obtained an Equal Error Rate of 0.05% over ZJU dataset and 0.5% over our dataset which shows that our model is superior to the state-of-the-art baselines.

Keywords: Gait verification. Time series classification. Binary classification. Pairwise loss function.

1 Introduction

Biometrics have been tremendously used for identity verification because of their high accuracy and low risk of breaching. Current biometric-based identity verification systems work using a variety of human features like fingerprint, iris or face, where user interaction is necessary, thus yielding a limited advantage over traditional security systems that require PIN or password. On the other hand, gait patterns are unique to every individual [SJ15a, WWP18], and can be identified without explicitly interacting with the person, making it a potential biometric option for identity verification.

Identity verification with gait, also called as gait verification, can be performed using three approaches: Video-based [Wa03], Floor based [Ve13] and Wearable sensors based [Ga07, SJ15a]. Video-based approaches work using a distant camera that records the individual's walk. In the floor based approach, sensor plates are fixed on the floor (called as force platforms), and walking patterns are identified through them. In the wearable sensor-based approach, sensors are placed on the human body at different locations to capture the gait motion.

In recent years, the latter approach became widely popular because of the advent in wearable technology. Unlike video-based approaches, this does not suffer from long-standing image processing problems like illumination variations, occlusion, clutter, etc. Moreover,

[1] Information Systems and Machine Learning Lab, University of Hildesheim, yalavarthi@ismll.de
[2] Information Systems and Machine Learning Lab, University of Hildesheim, josif@ismll.uni-hildesheim.de
[3] Computer Science Department, Hochschule Darmstadt, hareesh.mandalapu@h-da.de
[4] Information Systems and Machine Learning Lab, University of Hildesheim, schmidt-thieme@ismll.uni-hildesheim.de

it is very inexpensive compared to the floor based approaches. Inertial sensors (accelerometers, gyroscopes) required for gait recognition can be placed in clothes, shoes, belts, watches, and so on [SJ15b]. There are mobile applications which can collect the gait information through the accelerometer present in smartphones. Data produced by these sensors can be transmitted to the computer through wireless transmission technologies. One can develop models to process and analyze the data to identify or verify the user without her knowledge. It can be observed that user interaction is not necessary, which makes gait a potential future biometric security feature.

In the past, most researchers worked on gait recognition with accelerometer sensors by detecting the step cycles [Th11, Tr12], or signature points [Zh15]. Recently, Giorgi et al. [Gi18] developed a deep learning model with a recurrent neural network cell and the softmax cross-entropy loss function for gait recognition. Softmax cross-entropy like sigmoid cross entropy works well with the balanced data. However, in biometric applications, more specifically in biometric verification, data is highly imbalanced, and one needs special loss functions to handle the imbalance in the data.

In this paper, we present gait verification with deep learning architecture over two datasets. In the first dataset, called as ZJU dataset [Zh15], gait was recorded with accelerometer sensors placed on five different locations on the body for about 150 subjects. We also created a new dataset (see section 5) with 15 participants using the accelerometer sensor present in the smartphone held in the trousers' right pocket. We use pairwise loss function derived from the Bayesian Pairwise Ranking (BPR) [Re09] loss function for better classification. With extensive experimental evaluation, we show the effectiveness of our approach in identity verification with our model trained using pairwise loss function. The proposed model with pairwise loss function has an Equal Error Rate (EER) of 0.05%.

The contributions of this paper are as follows:

- We present deep learning model with a pairwise loss function for identity verification with gait.
- We develop a gait dataset with 15 subjects using accelerometer sensor present in the smartphone.
- Our empirical results show that with the proposed model using our pairwise loss function reduces the EER by 5 times over state-of-the-art algorithms.

2 Literature Review

Gait using accelerometers was first introduced by Morris [Mo04] for identity recognition and further formally addressed by Mantyjarvi et al. in [Ma05],[Ai05]. Since then, various techniques (see [SJ15b], [Zh15], [WWP18]) have been developed for analyzing the gait data collected from accelerometer (inertial) sensors. These techniques can be divided into two main approaches: signal matching based and machine learning-based.

Signal matching techniques try to match specific template/templates from gait signals [DMFM18] using a similarity measures like histogram similarity [GHS06], euclidean distance [GSB10], dynamic time wrapping [DMM17], cross-correlation [Ng14, Re15, SJ15b]

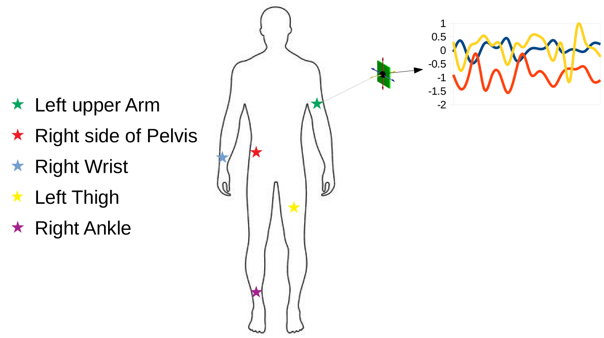


Figure 1: Body locations for sensors for ZJU dataset.

and Tanimoto similarity [Ng13, Su15]. The problem with these techniques is, they cannot perform well when the templates vary with the number of samples. To overcome this challenge, researchers develop segmentation algorithms to identify steps and cycles in the gait signal automatically. Researchers in [GSB10, Ju12, BS10] compared cycles by length normalization. These techniques suffer from inter-cycle phase misalignment. Researchers in [Tr12, Th11] used cycle comparison by alignment to overcome the problem of inter-cycle phase misalignment. However, dependency on cycle detection is a bottleneck in those methods. Zhang et al. [Zh15] addressed this problem by using salient points called as signature points (SPs). It is worth noting that they created one of the largest gait datasets with 175 subjects which we use for our experiments. Using their algorithm, Zhang et al. achieved an EER of 2.2% for identity verification. In our research, we use this dataset for the experimental evaluation.

Unlike signal matching techniques, machine learning-based techniques develop classification algorithms that can find the features in the signals and assign class labels to them. In this approach, the whole data is divided into training, and testing with training data is used to train the algorithm. Once the algorithm is trained, it can be readily implemented on any verification device. Nickel et al. [NBB11] collected walking dataset of 48 subjects using the accelerometer present in the smartphone. They extracted features like MFCC and BFCC, which are widely used for speech processing and employed support vector machines for the classification. They achieved a minimum False Acceptance Rate (FAR) of 5.9% and False Rejection Rate (FRR) of 6.3%. Recently, Giorgi et al. [Gi18] proposed a CNN model with recurrent neural network cell separately for every time series variable for the recognition and achieved a recognition rate of 97.5% over dataset created by Zhang et. al. The problem with this approach is that they could not be able to capture the features across the sensors, which result in low accuracy. Moreover, the usage of RNN cell for every time series variable is costly. Unlike them, our approach is very efficient as we use only one convolution layer and extract features across the variables making it more effective in terms of error reduction.

3 Background

In this section, we provide a brief introduction to gait, time series classification, and deep learning approach to it.

Gait: Gait is a human's manner of walking and is unique to everyone, making it a promising biometric. Gait can be easily reflected by the sensors (like accelerometers) placed on the specific locations of the human body and measured at three orthogonal directions X, Y, and Z. Unlike other biometrics using fingerprint, face or iris, gait can be easily recognized without human intervention. The advent of wearable technology helps in developing sensors that can be placed in shirts, trousers, watches, etc. Even, we find the accelerometer sensor that can record gait, in smartphones. These sensors provide the data uninterruptedly helping us to use gait biometric for continuously identify verification.

Time series classification: Classification of sequentially-ordered data measurements into their respective class labels is called time series classification. Data with the sequence of measurements collected over one variable is called as univariate time series data, whereas if it is measured over multiple variables, it is called multivariate time series data. In this work, gait is measured using accelerometer sensors with three orthogonal directions X, Y, and Z. Hence, we perform a multivariate time series classification for gait verification.

Deep Learning: Deep learning is a branch of machine learning which consists of cascades of multiple layers of non-linear processing units. Information is processed by feeding the output of one layer to the successive layer until it reaches the end of the network. Because of non-linear processing units, it has been widely applied in various applications like machine vision, time-series data analysis, robotics, and so on. Among the variety of deep learning models, convolution neural networks have attracted a wide range of researches because of their tremendous applications.

Convolutional Neural Network (CNN): CNN consists of multiple hidden layers, of which most of them are convolutional layers. Input is passed to a convolutional layer where the weights of the neurons in the layer are used to perform convolution operation on the information received. Generally, a non-linear activation function is applied to the output of the convolution layer and passed to the pooling layer (Average or Maximum). These steps can be performed multiple times: convolution layer followed by the pooling layer followed by another convolution layer. After processing the input by a series of convolutional and pooling layers, the obtained output is the smaller version of the input features. This output is fed to the fully connected network, which helps in training from the features extracted in the previous layer, while the convolutional layer helps in the representation of input sequences with reduced dimensions. This helps CNNs to emerge as a robust technique for various machine learning tasks like computer vision, reinforcement learning, time series analysis, to name a few. Figure 2 is an example of a CNN with one convolution layer, one max pooling layer, one average pooling layer, and one fully connected layer.

Training CNN: Training of CNN takes place in two stages: 1) Forward pass and 2) Backward pass. In the forward pass, input data is fed to CNN, and output is collected at the last layer. A loss function $\mathscr{L}$, is used to compute the error between the actual output (y) and predicted output ($\hat{y}$). Various kinds of loss function like logistic loss, sigmoid cross-entropy loss, softmax cross-entropy loss functions have been developed in the past among which sigmoid cross-entropy (SCE) loss functions are widely used for binary classification. In Section 4.2 we derive a pairwise loss function for binary classification.

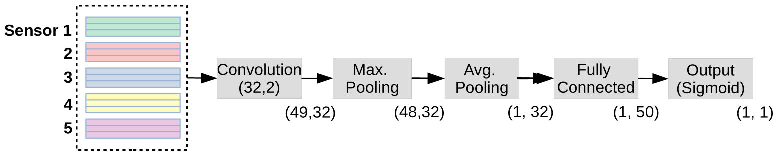


Figure 2: Proposed deep learning model.

4 Proposed Deep Learning Model

We consider gait verification as time series binary classification problem where the model results whether the input gait pattern belongs to a specific person. For this, we present deep learning architecture for the classification of Gait data.

4.1 Proposed model

We report the proposed model in figure 2. As accelerometer sensors measure data in 3 dimensions, we have multivariate time series data. We extract the features from the input data through a 1D convolutional layer with 32 kernels of size 2 and rectified linear unit (ReLU) activation function. We pass the output of the convolution layer through max-pooling, followed by the average pooling layers. Convolutional layer helps in extracting the features from input through local spatial coherence with a small receptive field. The max and average pooling layers help in reducing the dimensions of the extracted features and improve the efficiency of training. We pass these low dimensional features to a fully connected layer with 50 neurons. These layers help in learning from all the combinations of the low dimensional features collected previously. The fully connected layer is connected to one output neuron. We use ReLU activation function for the nodes present in the fully connected layer, whereas in the output node, we use a sigmoid activation function. We performed batch normalization after average pooling layer and fully connected layer. We use ADAM optimizer with a learning rate of 0.001.

We tune the hyperparameters like the number of epochs, batch size, learning rate, number of kernels, kernel size, number of neurons in a fully connected layer that are required for training, manually.

4.2 Pairwise loss function

We are dealing with the problem of biometric verification where the training data is highly imbalanced: very few positive instances and a large pool of negative instances, making the learning process difficult. The bulk of the literature addressed this problem of classification of imbalanced data by up-sampling the instances of a less appeared class or by downsampling the instances of frequently appeared class. The main drawback of these techniques is, either we get unwanted instances (during up-sampling) or miss the most important instances (during down-sampling) which hinders the performance of the algorithm. Unlike this approach, we take data as they are, but, sample them while computing the loss by giving more weight to the less appeared class sequences. For this, we develop a pairwise loss function with the desired characteristics to handle the imbalance in the data.

Consider binary classification problem for the dataset of $N = \{(x_i, y_i)_{i=1}^{N}\}$ instances. We consider instances belong to positive class as positive instances and of negative class as negative instances. $N^+ = (x_p, y_p) | p \in \{1, 2, 3, ..., N\} \wedge y_p = 1$ denotes set of positive samples whereas $N^- = (x_n, y_n) | n \in \{1, 2, 3, ..., N\} \wedge y_n = 0$ denotes set of negative samples

($N = N^+ \cup N^-$). Our aim is to group the input sequence x_i to its respective class using a CNN with one output node. The output of CNN for an input sequence x_i can be represented as $\hat{y}_i = \}(x_i, \phi)$ where ϕ represents network parameters. For evaluation purpose, we set $(x_i, \hat{y}_i) \in N^+$ if $\hat{y}_i \geq \theta$ where θ is a predetermined threshold value. Otherwise we set $(x_i, \hat{y}_i) \in N^-$. Consider, $\hat{y}_p$ and $\hat{y}_n$ are the outputs of the CNN for the p^{th} positive instance and the n^{th} negative instance respectively. Obviously, we want $\hat{y}_p \geq \theta > \hat{y}_n$.

This can be achieved by maximizing the difference between the outputs of positive and negative instances. The loss function $\mathscr{L}$ can be represented as:

$$\mathscr{L} = \frac{-1}{|N^+| \times |N^-|} \sum_{p=1}^{|N^+|} \sum_{n=1}^{|N^-|} (\hat{y}_p - \hat{y}_n) \tag{1}$$

In our pairwise loss function, we provide more weight on correctly predicting positive class compared to that of negative class as there are only a few instances of the former are available. This improves the performance of CNN model over the unbalanced dataset.

5 Dataset description and preprocessing

Here, we will briefly describe the datasets used in our study, and the preprocessing performed to make it ready for classification.

5.1 ZJU dataset:

This is a publicly available dataset created by Zhang et al. [Zh15]. This dataset contains gait patterns collected from 175 subjects in two different sessions. Out of 175 subjects, 153 people attended for both the sessions, and 22 subjects attended for only one. The time interval between two sessions for each subject varies from one week to six months. In each session, subjects are requested to walk on a level floor of 20m length with five accelerometer sensors mounted on their body at different locations: the left upper arm, the right wrist, the right side of the pelvis, the left thigh, and the right ankle as shown in figure 1. These sensors measure the acceleration in three directions (X, Y, and Z) simultaneously with a frequency up to 100Hz. For each subject, six recordings are taken in one session. There are 12 recordings for 153 subjects and six recordings for 22 subjects who attended for only one session. In this work, we considered the data of 153 subjects who attended both the sessions. A detailed description of the dataset can be found in [Zh15].

Preprocessing: The dataset is a multivariate time series dataset with 15 variables as explained above. Every sample has a different length ranging from 7 to 14 secs. However, for the training of CNN, all the inputs must be of the same dimensions. Hence, we have taken the data between 2 to 7 secs: there is some noise in the initial 2 secs (during the beginning of the walk), and everyone walked at least for 7 secs. Now we have 5 secs gait recording for all the data samples, which leads to 500 points per recording (recording frequency 100 Hz). Now to remove the noise in the gait cycle, we performed average sampling for a sample size of 10, which leads to data of 50 points per recording. Now our each sample has dimension 15×50. Once we reduce the noise from the data, we perform *data standardization*. Here we rescale the values of every variable in the time series data to 0 mean and

property	value
# Subjects	15
Sample frequency	100Hz
Participants age	24 - 55 Yrs
Female to male ratio	$\sim$ 1:6
# records	10
Floor length	40m
Length of recording	25 - 31s
# Step cycles in a record	38 - 40
Smartphone model	Samsung Galaxy J7 Pro
Mobile application used	Physics toolbox

Table 1: Basic statistics of our dataset.

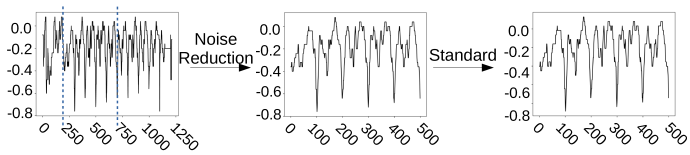


Figure 3: Prepossessing of ZJU dataset.

unit standard deviation. Standardization helps the observations to fit in the gaussian distribution. The sensor recordings for different sensors and subjects are on different scales, and standardization brings them to the same scale, thus improves the training quality. In Figure 3, we show the noise reduction and standardization performed in our study.

5.2 Our dataset

We created a new dataset with 15 participants holding a smartphone in their right front pocket of the trousers. Accelerometer present in the smartphone is used to record the gait using the software application *Physics toolbox* [4]. We asked the participants to walk naturally on the flat surface of 40m for 10 times in one session and recorded the data at a frequency of 100Hz. We present basic statistics of our dataset in Table 1.

preprocessing: Here, the data have three variables, which are the readings of the accelerometer in X, Y, and Z directions. Average walking time varies from 25 to 31 secs and considered the sequence between 5 to 20 secs for the experiments which is similar to the preprocessing performed over GJU-Accgait data (see Section 5.1). We also performed downsampling and standardization, as described earlier.

6 Experimental Results and Discussion

In this section, we present the experimental results on the Gait datasets described in section 5. We begin with the experimental setup.

[4]https://physics-toolbox-suite.en.uptodown.com/android

6.1 Experimental Setup

We evaluate the proposed model for the measure of Equal Error Rate (EER) with two kinds of setups, as mentioned below:

Same day. Here, we take the gait data of the subjects from one day only. For ZJU dataset, we take 4 samples for training and 2 for testing randomly. For our dataset, we experimented on one day. Hence, we consider it a same-day setting where we used 7 samples for training and 3 for testing.

Mixed day. Here, we take data of the subjects from both the days for ZJU dataset. We split 4 gait sequences of a subject per session for training and 2 for testing. On the whole, our training data has around 1224 training data samples and 612 testing data samples.

Competing algorithms.

(1) CNN with our pairwise loss (CNN+PW): This is our proposed CNN model with the pairwise loss explained in Section 4.2.

(2) CNN with sigmoid cross-entropy loss (CNN+SCE): This is our proposed CNN model with the widely used sigmoid cross-entropy loss.

(3) Support Vector Machines (SVM): We use the support vector machines presented in [NBB11] for gait recognition. We extracted the features of MFCC, bin distribution, minimum value, max value, and standard deviation for all the sensors along all the axes after standardization of the data.

(4) Recurrent CNN (RCNN): This is the model proposed in [Gi18] for gait recognition. It has two convolutional layers and bidirectional recurrent unit cell.

Computing threshold θ for EER

Here, we show computing the threshold value for competing algorithms used in this paper.

θ for pairwise (PW) loss: For the pairwise loss function, we have one output node for the proposed model with a sigmoid activation function. Let o denote the output of the output node. Input sequence is given positive label if $o \geq \theta$ and negative label if $o \leq \theta$. We vary θ from 0 to 1 to compute EER.

θ for Sigmoid Cross Entropy (SCE) Loss: For the SCE loss function, we have two output nodes with linear activation function. Let o_1 and o_2 denote output of the two nodes and, the label of the input sequence is positive (or negative) if $\sigma(o_1) - \sigma(o_2)$ is greater than (or less than) a given threshold θ. We vary θ from -1 to 1.

θ for Support Vector Machines (SVM): We implemented SVM using LinearSVM from SKlearn library [5] in python. It outputs the score for every input sequence, and we varied the score from -2 to 2 to compute the threshold for EER.

[5] www.scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

Setup	CNN+PW avg. EER	CNN+SCE avg. EER	SVM avg. EER	RCNN avg. EER
Mixed day	0.05 %	0.23 %	0.55 %	3.4 %
Same day-day1	0.1 %	0.22 %	0.65 %	3.3 %
Same day-day2	0.1 %	0.21 %	0.42 %	48.0 %
Our data	0.5 %	7.1 %	1.1 %	17 %

Table 2: EER scores for competing algorithms.

6.2 Experimental Results

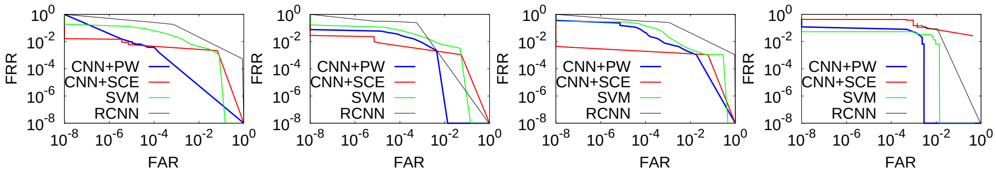
Here, we present the results for the experimental setup mentioned above.

EER comparison. We demonstrate the comparison of various algorithms mentioned above for computing the EER, for both mixed day and same day setups. We present the comparison results in Table 2. We report the average of EERs computed to verify all the subjects separately. We find that our model using PW loss outperforms all the other competing algorithms. As an example, for mixed day setup of ZJU dataset, our CNN+PW model has EER of 0.05% whereas CNN+SCE model has 0.23%, SVM has 0.55% and recurrent CNN has 3.4%. For the experiments on our data, we achieved EER of 0.5% with our CNN+PW model whereas CNN+SCE, SVM, and RCNN achieved EER of 7.1%, 1.1% and 17% respectively. It can be seen that RCNN performs worse than all the other algorithms. We observed that the model has a high number of network parameters that overfits the data leading to poor performance. *These results clearly show that our proposed model with PW loss performs superior (5 times improvement) in terms of EER.*

Detection Error Tradeoff (DET) curves: Performance of models in identity verification can be evaluated using DET curves, which shows the trade-off between FAR and FRR. In Figure 4, we present DET curves for both the datasets. Average FAR and FRR over all the subjects are presented in the graphs. It can be observed that over model CNN+PW performs very well for higher values of FAR. As an example, in ZJU dataset and mixed day setup, for FAR of 0.01, CNN+PW achieved FRR of around 10^{-5}, whereas CNN+SCE, SVM and RCNN achieved around 10^{-2}, 10^{-2} and 10^{-1} respectively. For the lower values of FAR, CNN+SCE wins over our CNN+PW by a slight margin in ZJU dataset. For our dataset, SVM performs slightly better than our CNN+PW.

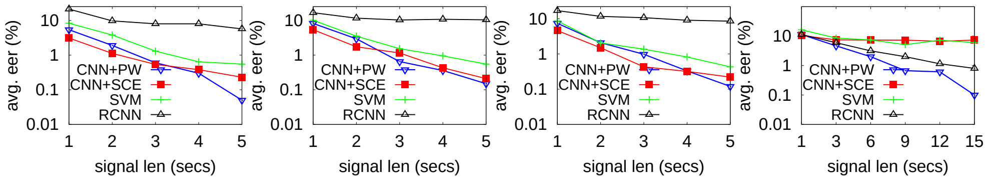
Comparison by varying signal length: We present the performance of the algorithms by varying the signal length as one may be interested in understanding the performance of the verification system with short signal length. Moreover, people do not want to wait for a long time to get verified. In the ZJU dataset, for all the experiments, we used the signal length of 5 secs as this is the maximum length common to every signal of sensors for all the subjects. For this, we varied signal length from 1 sec to 5 secs and computed the EER for both mixed and same day setups.

Furthermore, in our dataset, we varied the signal length from 1 sec to 15 secs. We present our results for varying signal lengths in Figure 5 where X-axis represents the signal length in seconds, and Y-axis represents average EER. Our CNN+PW outperforms all the other competitors over our dataset and all the setups over ZJU dataset. As an example, for single



(a) *DET curves: mixed* (b) *DET curves: day1* (c) *DET curves: day2* (d) *DET curves: Our data*

Figure 4: Detection Error Tradeoff (DET) curves of varying False Rejection Rate (FRR) Vs False Acceptance Rate (FAR) for different algorithms.



(a) *Var. signal len.: mixed* (b) *Var. signal len.: day1* (c) *Var. signal len.: day2* (d) *Var. signal len.: our data*

Figure 5: Comparing average Equal Error Rate (EER) with varying signal length (secs) for different algorithms.

day setup in ZJU dataset, we achieved an EER of 0.65% even for a signal length of 3 secs. Whereas for CNN+SCE reached an EER of 1.15%, SVM and RCNN reached EER of 1.53% and 10.30% respectively. *Clearly, with our model we can reach reasonable EER even with less signal length.*

Sensor Location	Avg. EER %	Sensor Location	Avg. EER %	Sensor Location	Avg. EER %
RW	3.12	RW, LA, RP	0.29	RW, LT, RA	0.42
LA	4.13	RW, LA, LT	0.32	LA, RP, LT	**0.16**
RP	**2.54**	RW, LA, RA	0.33	LA, RP, RA	0.21
LT	4.66	RW, RP, LT	0.18	LA, LT, RA	0.19
RA	4.55	RW, RP, RA	0.48	RP, LT, RA	0.17
RW, LA	0.59	LA, LT	0.49	RW, LA, RP, LT	0.16
RW, RP	0.63	LA, RA	0.53	RW, LA, RP, RA	0.15
RW, LT	0.62	RP, LT	**0.51**	RW LA, LT, RA	0.11
RW, RA	0.68	RP, RA	0.71	RW, RP, LT, RA	0.13
LA, RP	0.72	LT, RA	0.74	LA, RP, LT, RA	**0.07**
RW, LA, RP, LT, RA					**0.05**
RW - Right Wrist, LA - Left upper Arm, RP - Right side of Pelvis, LT - Left Thigh, RA - Right Ankle					

Table 3: Sensor filtering using CNN+PW model

Sensor filtering. ZJU dataset is created by taking the data from the sensors placed at 5 locations on the body. In this section, we evaluate the capacity of these sensors for the gait classification. We show the EERs for all the combinations of the sensors in table 3 with our CNN+PW model. Performance of the model increases with an increase in the number of sensors used. For the data with a single sensor, we achieved the best EER of 2.54% for sensor placed on the right side of the pelvis. For the combinations of 2 $\sim$ 4 sensors, we achieved EER of 0.51% $\sim$ 0.07%. We achieved 0.05% of EER for the data with all the sensors. We observed that the sensor placed on the right side of pelvis plays a key role in the gait verification compared to the sensors on other locations.

7 Conclusion

In this paper, we present biometric verification using gait measured with accelerometer sensors. We formulated the problem as a time-series binary classification. We employed deep learning architecture with a pairwise loss for the binary classification which does not require detecting cycles or feature extraction from the signal. Based on a detailed experimental evaluation over two gait sets of subjects 15 and 150, our architecture reduces Equal Error Rate (EER) by 5 times compared to state-of-the-art techniques.

Acknowledgements

This work is co-funded by the industry project "Data-driven Mobility Services" of ISMLL and Volkswagen Financial Services.https://www.ismll.uni-hildesheim.de/projekte/dna_en.html)

References

[Ai05] Ailisto, Heikki J; Lindholm, Mikko; Mantyjarvi, Jani; Vildjiounaite, Elena; Makela, Satu-Marja: Identifying people from gait pattern with accelerometers. In: Biometric Technology for Human Identification II. volume 5779. International Society for Optics and Photonics, pp. 7–15, 2005.

[BS10] Bours, Patrick; Shrestha, Raju: Eigensteps: A giant leap for gait recognition. In: 2010 2nd International Workshop on Security and Communication Networks, IWSCN 2010. pp. 1 – 6, 06 2010.

[DMFM18] De Marsico, Maria; Fartade, Eduard Gabriel; Mecca, Alessio: Feature-based Analysis of Gait Signals for Biometric Recognition. 2018.

[DMM17] De Marsico, Maria; Mecca, Alessio: Biometric walk recognizer. Multimedia Tools and Applications, 76(4):4713–4745, 2017.

[Ga07] Gafurov, Davrondzhon: A survey of biometric gait recognition: Approaches, security and challenges. In: Annual Norwegian computer science conference. pp. 19–21, 2007.

[GHS06] Gafurov, Davrondzhon; Helkala, Kirsi; Søndrol, Torkjel: Biometric Gait Authentication Using Accelerometer Sensor. JCP, 1(7):51–59, 2006.

[Gi18] Giorgi, Giacomo; Martinelli, Fabio; Saracino, Andrea; Sheikhalishahi, Mina: Walking Through the Deep: Gait Analysis for User Authentication Through Deep Learning. In: 33th IFIP International Conference on ICT Systems Security and Privacy Protection (SEC). volume AICT-529, Poznan, Poland, pp. 62–76, September 2018.

[GSB10] Gafurov, Davrondzhon; Snekkenes, Einar; Bours, Patrick: Improved gait recognition performance using cycle matching. In: 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops. IEEE, pp. 836–841, 2010.

[Ju12] Juefei-Xu, Felix; Bhagavatula, Chandrasekhar; Jaech, Aaron; Prasad, Unni; Savvides, Marios: Gait-id on the move: Pace independent human identification using cell phone accelerometer dynamics. In: 2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS). IEEE, pp. 8–15, 2012.

[Ma05] Mantyjarvi, Jani; Lindholm, Mikko; Vildjiounaite, Elena; Makela, S-M; Ailisto, HA: Identifying users of portable devices from gait pattern with accelerometers. In: Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005. volume 2. IEEE, pp. ii–973, 2005.

[Mo04] Morris, Stacy J: A shoe-integrated sensor system for wireless gait analysis and real-time therapeutic feedback. 2004.

[NBB11] Nickel, Claudia; Brandt, Holger; Busch, Christoph: Classification of acceleration data for biometric gait recognition on mobile devices. BIOSIG, 2011.

[Ng13] Ngo, Thanh Trung; Makihara, Yasushi; Nagahara, Hajime; Mukaigawa, Yasuhiro; Yagi, Yasushi: The largest inertial sensor-based gait database and performance evaluation of gait-based personal authentication. hip, 1:2, 2013.

[Ng14] Ngo, Trung Thanh; Makihara, Yasushi; Nagahara, Hajime; Mukaigawa, Yasuhiro; Yagi, Yasushi: Orientation-compensative signal registration for owner authentication using an accelerometer. IEICE TRANSACTIONS on Information and Systems, 97(3):541–553, 2014.

[Re09] Rendle, Steffen; Freudenthaler, Christoph; Gantner, Zeno; Schmidt-Thieme, Lars: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence. AUAI Press, pp. 452–461, 2009.

[Re15] Ren, Yanzhi; Chen, Yingying; Chuah, Mooi Choo; Yang, Jie: User verification leveraging gait recognition for smartphone enabled mobile healthcare systems. IEEE Transactions on Mobile Computing, 14(9):1961–1974, 2015.

[SJ15a] Sprager, Sebastijan; Juric, Matjaz: Inertial sensor-based gait recognition: a review. Sensors, 15(9):22089–22127, 2015.

[SJ15b] Sprager, Sebastijan; Juric, Matjaz B: An efficient HOS-based gait authentication of accelerometer data. IEEE transactions on information forensics and security, 2015.

[Su15] Subramanian, Ravichandran; Sarkar, Sudeep; Labrador, Miguel; Contino, Kristina; Eggert, Christopher; Javed, Omar; Zhu, Jiejie; Cheng, Hui: Orientation invariant gait matching algorithm based on the Kabsch alignment. In: IEEE International Conference on Identity, Security and Behavior Analysis (ISBA 2015). IEEE, pp. 1–8, 2015.

[Th11] Thanh Trung, Ngo; Makihara, Yasushi; Nagahara, Hajime; Sagawa, Ryusuke; Mukaigawa, Yasuhiro; Yagi, Yasushi: Phase Registration in a Gallery Improving Gait Authentication. 2011 International Joint Conference on Biometrics, IJCB 2011, 10 2011.

[Tr12] Trung, Ngo Thanh; Makihara, Yasushi; Nagahara, Hajime; Mukaigawa, Yasuhiro; Yagi, Yasushi: Performance evaluation of gait recognition using the largest inertial sensor-based gait database. In: 5th IAPR International Conference on Biometrics (ICB). 2012.

[Ve13] Vera-Rodriguez, Ruben; Mason, John S.D.; Fierrez, Julian; Ortega-Garcia, Javier: Comparative Analysis and Fusion of Spatiotemporal Information for Footstep Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013.

[Wa03] Wang, Liang; Tan, Tieniu; Ning, Huazhong; Hu, Weiming: Silhouette analysis-based gait recognition for human identification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(12):1505–1518, Dec 2003.

[WWP18] Wan, Changsheng; Wang, Li; Phoha, Vir V: A survey on gait recognition. ACM Computing Surveys (CSUR), 51(5):89, 2018.

[Zh15] Zhang, Yuting; Pan, Gang; Jia, Kui; Lu, Minlong; Wang, Yueming; Wu, Zhaohui: Accelerometer-Based Gait Recognition by Sparse Representation of Signature Points With Clusters. IEEE Transactions on Cybernetics, 45(9):1864–1875, Sep. 2015.