IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 2, 2019, accepted January 6, 2020, date of publication January 13, 2020, date of current version January 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2966142

Data Augmentation for Inertial Sensor-Based Gait Deep Neural Network

LAM TRAN[1,2] **AND DEOKJAI CHOI**[2]

[1]Information Technology Faculty, Ho Chi Minh University of Sciences, Ho Chi Minh city 700000, Vietnam
[2]Electronics and Computer Engineering School, Chonnam National University, Gwangju 61186, South Korea

Corresponding author: Lam Tran (thlam@fit.hcmus.edu.vn)

This work was supported by the Korea under Grant NRF-2017R1D1A 1B03035343.

ABSTRACT Inertial sensor-based gait has been discovered as an attractive method for user recognition. Recently, with the approaching of deep learning techniques, new state-of-the-art researches have been established. However, the scarcity of training data still endures as an obstacle that impedes to build a robust deep gait model. In this study, we address that problem by proposing a novel approach for inertial sensor-based gait data augmentation. First, two label-preserving transformation algorithms, namely Arbitrary Time Deformation (ATD) and Stochastic Magnitude Perturbation (SMP), are proposed to generate more training data from the real gait data. The ATD algorithm adjusts the timing information of gait data with random values, on the other hand, SMP alters the magnitude arbitrarily, to create variations on the augmenting data. Then, we design a generic gait recognition model using convolutional neural network, in which, the ATD and SMP algorithms are coordinated appropriately to produce augmenting data varied naturally in both time and magnitude as real data. The proposed approach was evaluated on two public datasets, one was collected in unconstrained conditions, and the other had the largest number of participating users. The experiment showed that, under different amounts of training data, using ATD or SMP alone could increase the recognition performance effectively, and their combination even attained higher accuracy. With ATD and SMP, our model achieved competitive performance on both two datasets comparing to state-of-the-art researches.

INDEX TERMS Gait recognition, inertial sensor data, accelerometer data, CNN gait network, data augmentation.

I. INTRODUCTION

Gait recognition refers to the task of verifying or identifying humans by their walking pattern. Traditional researches in this field mainly base on computer vision [1], [2], or floor sensor technologies [3], [4], which are useful for the context of video surveillance or security access control in restricted areas. The development of micro-machining technique has opened a new gait recognition approach, which uses inertial sensors (*e.g.,* accelerometer, gyroscope) attached to the human body to capture the walking pattern [5]. As offering various attractive advantages (*e.g.,* small-size, lightweight, mobility, low-cost, implicit operation), this approach has attracted significant attention from researchers worldwide [6]–[12], and achieved promising results. Despite those merits, most gait recognition researches were conducted in laboratory environment, under certain operation constraints, and are infeasible for practical applications [7], [13].

Although deep learning techniques have been applied and established new state-of-the-art researches in this field recently [9]–[12], [14]–[16], the scarcity of training data still endures as an obstacle that impedes to build a robust deep gait model [7]. Specifically, training a deep network usually requires a large and diverged dataset which covers a substantial number of data instances possibly occurring in practice [17], [18]. However, most of the available inertial sensor-based gait datasets could not satisfy such requirements. For instance, the published datasets which were collected under realistic conditions, just contain data of a limited number of users (*e.g.,* 20 users [19], 38 users [20], 51 users [21]). On the other hand, the dataset which has a large user population (*i.e.,* 744 users), has limited data instances for each user, and was collected in a constrained environment

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Kamrul Hasan.

(*i.e.,* each user performed two level-walk sequences of 9 m) [22]. Typically, collecting a dataset satisfying the above requirements is a difficult and high-cost task. Thus, it is desired to have a practical and effective method to obtain more training data without performing the data acquisition process.

Data augmentation by label-preserving transformation is an elegant strategy to address the data scarcity problem. In this strategy, the amount of training data is increased multiple times by introducing noise or artificially modifying the available data so that they are different but still represent a same identity [23], [24]. Data augmentation has showed its effectiveness on various machine learning tasks (*e.g.,* computer vision [23]–[27], speech and language processing [28]–[30]). Unfortunately, to the best of our knowledge, there is no published study that proposes and analyzes a data augmentation method dedicated to gait. In addition, although there are substantial data augmentation methods for other data types (*e.g.,* image, voice), they could not be adopted for gait as each data type has a different structure, meaning and variation pattern.

Data augmentation is a promising approach to enhance the robustness of inertial sensor-based deep gait model. However, to effectively improve the performance, the augmentation method needs to meet the following requirements. First, the augmenting data should have certain variations to be relatively different from the real data instance that it is derived from. Second, the generating data should inherit the crucial information from the real data so that they both represent a same identity. Third, the variations introducing in augmenting data should be highly possible to happen in practice so that the trained model will be more robust in real environments. Final, the data augmentation method should be applicable to different sensor models, regardless of their specifications (*e.g.,* sampling rate, sensitivity). Specifically, as having different settings, each sensor may have its own range value and variation. The data augmentation method should be able to detect such properties and generate augmenting data accordingly.

Motivating by the promising and necessity of the work, in this study, we propose a novel method for inertial sensor-based gait data augmentation. Our main ideas to fulfill the aforementioned requirements and effectively improve the performance are explained as follows. We observed that: *(i)* the gait data variation could be decomposed into two domains as time and magnitude; and *(ii)* the variation of real data is usually resulted from the combination of variations in these two domains (see Section III-B.1). So, we proposed two algorithms, namely *Arbitrary Time Deformation (ATD)* and *Stochastic Magnitude Perturbation (SMP)*, to create random variations in the time and magnitude domains, respectively. Intuitively, ATD plays a role as the outside factors which cause variation in timing information (*e.g.,* human walking speed, unstable sampling rate of the sensor). On the other hand, SMP is used to simulate the impact of those factors affecting the magnitude (*e.g.,* walking surface, footwear, clothing). Both of them are designed so that the variations are added randomly to make the augmentation data arbitrarily different from the real data, however, the critical information of real gait data is preserved on the augmenting data. In addition, gait data acquired in real conditions is always affected by various factors and varies in both time and magnitude domains. So, we design a generic deep gait recognition model based on the convolutional neural network (CNN), in which, ATD and SMP are coordinated effectively to achieve optimal performance. In summary, the main contributions of our study are outlined as follows:

- First, by observing how the real gait data vary in practice, we proposed two data augmentation algorithms as ATD and SMP to address the data scarcity problem and improve the robustness of deep gait model (Section III-B). To the best of our knowledge, there was no study addressing this task before.
- Second, we designed a generic deep CNN gait model for gait identification, in which the proposed augmentation algorithms were coordinated appropriately to improve classification performance (Section III).
- Third, we extended the proposed identification gait model to construct the verification model using transfer learning and one-class support vector machine (OCSVM) [31] (Section IV-D.1). By that, we aimed to additionally confirm the effectiveness of augmentation algorithms on the verification task.
- Fourth, we constructed a comprehensive experiment to evaluate the proposed methods using the realistic dataset (*i.e.,* Chonnam National University gait dataset (CNU) [20]) and the largest gait dataset (*i.e.,* Osaka University gait dataset (OU-ISIR) [22]) (Section IV). By CNU and OU-ISIR, we analyzed our proposed methods under realistic conditions and large user population, respectively. Moreover, as they were captured using different sensor models, we demonstrated that our algorithms can operate effectively on different sensor specifications. With the proposed data augmentation algorithms, our model performance was significantly enhanced and achieved competitive results (*i.e.,* for CNU dataset, our model achieved the identification accuracy (ACC) of 96.24 %, and the equal error rate (EER) on verification task of 1.62 %; for OU-ISIR dataset, it achieved 89.99% ACC and 4.49% EER).
- Last, we provided a detailed analysis of the proposed data augmentation algorithms to demonstrate their effectiveness under different amounts of real training data and augmenting data. In addition, the instruction on how to select appropriate settings for the augmentation algorithms to achieve optimal performance was also given.

II. RELATED WORK

In this section, we present a review of state-of-the-art researches related to our study, which are divided into two groups as follows.

A. GAIT RECOGNITION

For a long time, human gait has been discovered to contain individual discriminative information, which can be used to identify or verify the owner [32], [33]. By the data acquiring techniques, the gait recognition researches could be categorized into three approaches: computer vision, floor sensor, and inertial-sensor [13]. The computer vision approach uses the user's video (while he/she is walking) as the data source for recognition [1], [2]. This approach is potential for the context of video surveillance, security access control in restricted areas, or early detection of interest person. The floor sensing-based method uses the signals sampling from embedded sensors in the smart mat, or floor vibration sensors to identify the walking user [3], [4]. This approach is useful to the task of transparently recognizing people when they enter a specific area.

Along with the development of technology, inertial sensors are increasingly adopted in many practical applications, and widely embedded in mobile/wearable devices (*e.g.,* smartphone, smartwatch, inertial measurement unit (IMU)). This technology offers a new gait recognition approach, in which, the signal data sampling by inertial sensors attached to the user's body (while he/she is walking) is used as data source for user recognition [7], [13]. The first gait recognition study using inertial sensor was conducted by Ailisto *et al.* [5]. Offering many attractive advantages (*e.g.,* small size, mobility, low cost, implicit operation), this approach quickly received significant research effort after that, and achieved promising results [6]–[8], [14]–[16], [34]. In the beginning stage, conventional machine learning and pattern recognition techniques were adopted to discover from raw data the meaningful feature/template for user identification/verification. The study [7] provides a detailed and sufficient summarization for all researches following such approaches.

Recently, deep learning has gained an extraordinary development and dramatically improved the state-of-the-art researches in many pattern recognition and machine learning tasks such as speech recognition, visual object recognition/detection [35]. Following that trend, many studies have adopted deep leaning techniques for the task of inertial sensor-based gait recognition and achieved new state-of-the-art results [9]–[12], [14]–[16], [36].

Specifically, Gadaleta and Rossi [10] were the volunteers who conducted the first CNN framework named IDNet for smartphone-based gait recognition. In that research, the accelerometer and gyroscope signals were segmented into gait cycles, following by the orientation-invariant transformation and normalization to get the fixed-length gait cycles. The gait cycle was used as the input of CNN to extract the deep gait features which were then used for predicting user identity. The IDNet was evaluated on a dataset of 50 subjects and achieved a very promising performance as 0.15% equal error rate (EER). After that, several studies [14], [36] followed this trend by proposing others CNN architectures, and evaluated with different datasets (*e.g.,* [19], [22]). On the other hand, some researches adopted CNN on the fused data of multiple sensors placing in different positions of the human body to improve the accuracy [9], [11]. Instead of directly using gait signals as the input of CNN, Zhao and Zhou [12] transformed gait signals to angle embedded gait dynamic image (AE-GDI), which is invariant to device's rotation or disorientation. The CNN then used AE-GDI image as the input for predicting the owner's identity. On the newest research [16], recurrent neural network (RNN) was firstly adopted on this task, and a promising result was reported.

B. DATA AUGMENTATION

Data augmentation is an elegant and effective approach to address the data scarcity problem and improve the robustness of deep learning model. In this approach, the amount of training data is increased multiple times by introducing noise or appropriately modifying the original data [23], [24]. In deep learning, with more training data, we can reduce the over-fitting, and train a more robust deep model. Due to its effectiveness, data augmentation has been widely adopted in various tasks (*e.g.,* image [23]–[27], speech recognition and language processing [28]–[30]).

Despite its promising, data augmentation has not been received proper research attention in inertial sensor-based gait recognition. To the best of our knowledge, there is no published research that focuses on proposing and analyzing a data augmentation method dedicated to inertial sensor-based gait data. There were only some simple methods, which were included in [9], [14]. In [9], each gait signal was diversified using a value drawn from a uniform distribution in the range of $[-0.2, 0.2]$ to make an augmenting instance. The method described in [14] generated new gait sequences by three steps: firstly, adding Gaussian noise with $\sigma = 0.01$ to each original signal; then, scaling the length to a random value in the range of $[0.7, 1.1]$; finally, interpolating to add more values. Despite their effectiveness, they lost the generality. Specifically, these methods are specific for the adopted sensor models. Each sensor type has its own sensitivity and operating range value specifying by the manufacturer. Then, to adopt these methods to data collected by other sensor models, such settings need to be re-identified. In addition, walking speed of a human is unstable. Thus, the nature gait signal usually varies in both the magnitude and acquiring time (Section III-B.1), which can not be simulated by the methods in [9], [14]. In summary, proposing a generic inertial sensor-based gait data augmentation method, and providing its detailed analysis are still the uncompleted tasks.

III. INERTIAL SENSOR-BASED GAIT RECOGNITION MODEL

In this section, we describe the inertial sensor-based gait identification model using convolutional neural network, in which, the proposed data augmentation algorithms are employed to improve the network performance. Here, we provide an overview of the model. The details of each processing step will be explained after that.

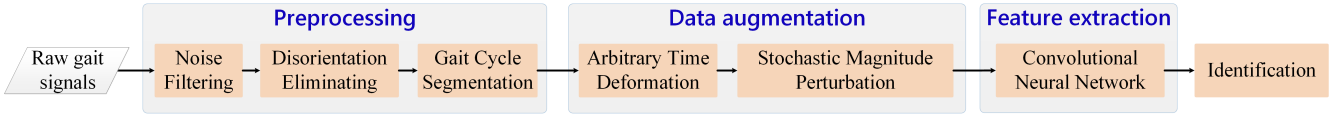


FIGURE 1. The processing blocks of the inertial sensor-based gait identification model using deep convolutional neural network, in which the data augmentation algorithms are employed to improve the performance.

Overview: The overall processing of the proposed gait identification model is depicted in Figure 1. Given raw gait data captured by the acceleration sensor, some preprocessing steps are performed to eliminate noise and impact of device's orientation instability. Then, the gait signals sequence are segmented into a set of *gait cycles*. For easy discriminating, we call the cycle segmented by this step as *real gait cycle*. After that, the ATD algorithm is adopted to generate more training gait cycles varying in time domain. The real gait cycles, along with the output of ATD algorithm, are used as input of the SMP algorithm, to obtain the data varying in magnitude, and in both time and magnitude, respectively. All the augmenting gait cycles, and the real cycles, are used for training a deep CNN to extract deep features, which are then used for user classification using fully connected network.

Note that, the state-of-the-art researches show that, the gyroscope data could be used along with the accelerometer data to improve the recognition performance [10], [14], [16]. However, in this study, we focus on proposing and analyzing a data augmentation approach. In addition, the processing techniques of gyroscope data are almost similar to those of accelerometer data. Thus, we just use the accelerometer data for the whole model, an extension that uses gyroscope data could be easy to find in related researches.

A. GAIT DATA PREPROCESSING

For data preprocessing and gait cycle extraction, we utilize the methods proposed in [20], which are briefly summarized as follows.

1) NOISE AND IMPACT OF DISORIENTATION ELIMINATING

The main data for using are the acceleration signals sampling by accelerometer. Typically, an acceleration signal acting on the sensor at the time t is represented by a 3-dimensional vector $\mathbf{a} = \left[a^X\ a^Y\ a^Z\right]^\top \in \mathbb{R}^3$, where a^X, a^Y and a^Z are the forces acting on the sensor in the X, Y and Z axes, respectively (see [20]). The raw acceleration data comprise a sequence of signals $\left\{\mathbf{a}_1, \mathbf{a}_2, \ldots \mathbf{a}_s\right\}$ sampling on sensor respectively at the time $\left\{t_1, t_2, \ldots t_s\right\}$, where s is the number of signals.

The first step of preprocessing is noise filtering, in which, the Daubechies orthogonal wavelet with the decomposition at level 2 [37] is used to clarify the noise caused from the data acquisition process. Then, the problem of device's orientation instability is addressed by leveraging the orientation sensor, which is widely embedded in most of mobile/wearable devices. This sensor provides the rotation angles between coordinate systems of the device and the Earth. A signal sampling from the orientation sensor is also represented by a 3-dimensional vector $\mathbf{o} = \left[\alpha\ \beta\ \gamma\right]^\top$, where α, β and γ are the rotation degrees along the axes of X, Y and Z, respectively. Similar to accelerometer, the orientation sensor also outputs a sequence of samples $\left\{\mathbf{o}_1, \mathbf{o}_2, \ldots \mathbf{o}_s\right\}$, which provide the device rotation states at the time $\left\{t_1, t_2, \ldots t_s\right\}$, respectively. For each acceleration signal $\mathbf{a}_j$ $(1 \leq j \leq s)$, we transform it from the device's coordinate system, which is unstable, to the Earth's coordinate system, using the orientation state $\mathbf{o}_j$ as explained in [20]. By this transformation, all acceleration signals are represented in a fixed coordinate system, and, the impact of device disorientation on the acceleration signal is eliminated.

2) GAIT CYCLE SEGMENTATION

In the next step, the acceleration gait signals are split into segments based on the walking cycle, where each segment (called *gait cycle*) comprises of signals acquired during the time of two consecutive normal walking steps. For this task, we use the methods proposed in [20]. First, we determine the negative peaks, which are those signals having the Z-axis magnitude significantly smaller than the mean of all signals measured on the Z-axis. Intuitively, the negative peaks are those signals potentially captured at the time when the leg touches the ground. Next, the average gait cycle length (*i.e.,* the number of signals in each gait cycle) is estimated using the auto-correlation. Then, with each pair of consecutive negative peaks, if the distance between them is approximate to the gait cycle length up to a pre-defined factor, the signals between them are extracted as a gait cycle. Figure 2 illustrates the acceleration signals in three axes after noise filtering and disorientation elimination, where the red verticals denote the delimitations of gait cycles detected using the described method.

Assume that there are m gait cycles extracted from a user. With each gait cycle i $(1 \leq i \leq m)$, we represent it by a $3 \times n_i$ matrix $\mathbf{A}_i$, where n_i is the number of signals. We mean $a^X_{i,j}$, $a^Y_{i,j}$, $a^Z_{i,j}$ as the magnitude of signal j $(1 \leq j \leq n_i)$ in $\mathbf{A}_i$, acted on the X, Y and Z axes, respectively. In addition, with each gait cycle $\mathbf{A}_i$, we represent the timestamp of its signals by a vector

$$\mathbf{t}_i = \left[t_{i,1}\ t_{i,2}\ \ldots\ t_{i,n_i}\right], \tag{1}$$

where $t_{i,j}$ is the timestamp of the signal j in cycle i.

B. GAIT DATA AUGMENTATION

In this section, we describe our approach for inertial sensor-based gait data augmentation. First, we show how the

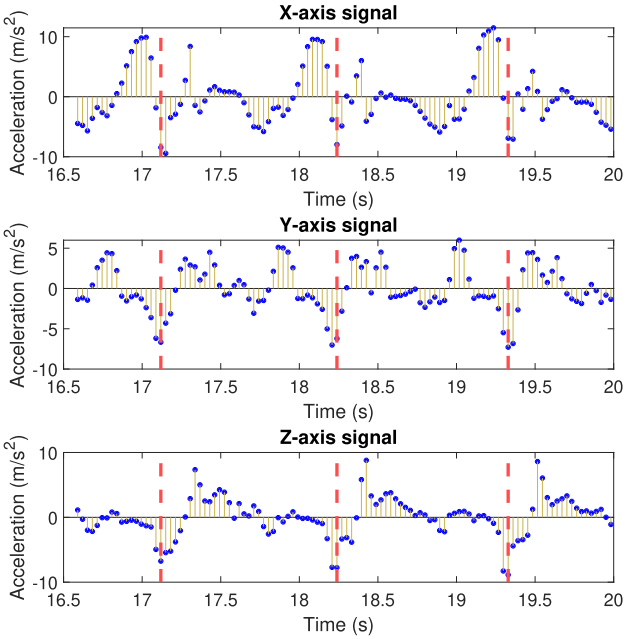


FIGURE 2. An example of the discrete acceleration signals in *X*, *Y* and *Z* axes captured while the user is walking, where the red verticals indicate the gait cycles' delimitations determined in the *Z* axis.

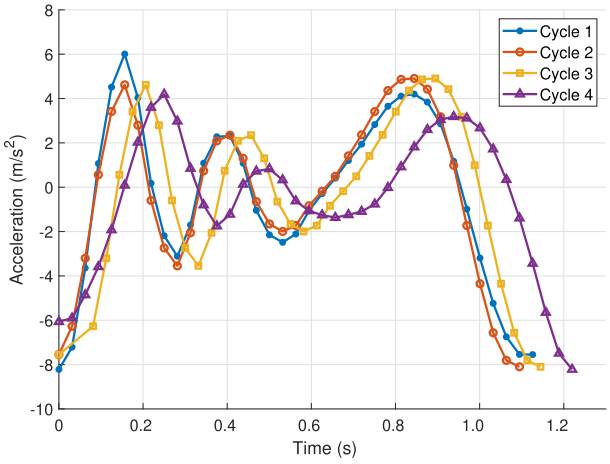


FIGURE 3. Acceleration signals of different gait cycles of a same user, capturing in *X*-axis.

signals in a gait cycle fluctuate when being acquired in real conditions. Then, we propose two data augmentation algorithms, which generate augmenting gait cycles by following the fluctuation styles of the real gait cycle.

1) VARIATION OF GAIT ACCELERATION SIGNALS

Figure 3 displays the X-axis acceleration signals of different gait cycles captured from a same user. We can see that the gait signals between different cycles are not identical but greatly vary from each other. This variation is caused by the difference in acquiring conditions such as walking speed, road surface condition, wearing clothes, footwear, backpack [7]. Here, we only illustrate the variation in X axis, actually, it occurs in both three axes X, Y and Z. It seems that the variations are chaotic, however, we can decompose them into two components as *time-based* and *magnitude-based*.

In the time-based variation, the gait signals of different cycles have the same (or approximate) magnitude values, however, their acting time (with respect to the starting signal of each cycle) is different. So, the time interval between two consecutive gait signals is inconstant, and the time length of each gait cycle is not identical but fluctuates in a specific range value. The factor causing this type of variation mostly is the instability of human walking speed. The gait cycles 2 and 3 in Figure 3 give an example of this variation. Although the magnitude of each signal of two gait cycles is approximate to each other, the signals in gait cycle 3 are captured 'slower' than those of cycle 2, and the length of cycle 3 (in time) is longer than cycle 2.

On the other hand, in magnitude-based variation, the gait signals of different cycles are captured at the same (or approximate) time, based on the starting time of each cycle. However, they have different magnitude values. The gait cycles 1 and 2 illustrate this kind of variation. This variation could be caused by various factors such as the differences in walking surface, wearing clothes or footwear.

In practice, the capturing gait signals are affected by multiples factors simultaneously. Thus, the variation of gait signals is usually resulted from the combined effects of magnitude-based and time-based variations. The gait cycle 4 in Figure 3 is an example for this type. Comparing to signals of other cycles, the signals of cycle 4 vary in both magnitude and acting time, however, the cycle 4 still maintains a similar waveform with other cycles.

2) THE ARBITRARY TIME DEFORMATION

Observing the time-based variation, we propose the *Arbitrary Time Deformation (ATD)* algorithm, which generates the augmenting gait cycles by introducing random variations on the timing domain of the real gait cycles. We design the ATD algorithm so that, not only the time length of gait cycle (*i.e.,* the time gap between the last and first signal in each gait cycle) varies, but also the time gap between each pair of consecutive signals fluctuates arbitrarily.

Given m real gait cycles of a user (see Section III-A.2), we determine their time length variance δ_τ^2 by

$$\delta_\tau^2 = \frac{1}{m-1}\sum_{i=1}^{m}(t_{i,n_i} - t_{i,1} - \bar{\tau})^2, \tag{2}$$

where n_i is the number of signal in the cycle i, and $\bar{\tau}$ is mean of the time length

$$\bar{\tau} = \frac{1}{m}\sum_{i=1}^{m}(t_{i,n_i} - t_{i,1}). \tag{3}$$

Next, the averaged number of signals per gait cycle is also determined,

$$\bar{n} = \frac{1}{m}\sum_{i=1}^{m} n_i. \tag{4}$$

After that, the ATD-based augmenting gait cycle is generated by altering the timestamps and copying the magnitude of the real gait cycle. Specifically, let $\tilde{\mathbf{t}} = [\tilde{t}_1\ \tilde{t}_2\ \ldots\ \tilde{t}_{n_i}]$ be the timestamps of an augmenting gait cycle which is generated from a gait cycle i. At first, $\tilde{\mathbf{t}}$ is initialized as the timestamps of its based gait cycle, $\tilde{\mathbf{t}} \leftarrow \mathbf{t}_i$. Then, the timestamp of each signal j ($2 \le j \le n_i$) in the new cycle is adjusted one-by-one with random value. When adjusting the timestamp of a signal j, we aim to alter the time gap between the signals j and $(j-1)$, however, preserve the time gap between j and all signals after j. So, at the step of adjusting signal j, the timestamp of all signals k ($j \le k \le n_i$) will be adjusted by a value Δt,

$$\tilde{t}_k \leftarrow \tilde{t}_k + \Delta t, \tag{5}$$

where Δt is a random number determined by

$$\Delta t = x \sim \mathcal{N}(0, \frac{\delta_\tau^2}{\bar{n}}) \mid x + \tilde{t}_j > \tilde{t}_{j-1}. \tag{6}$$

That is, timestamp of all signals from j to the last are adjusted by Δt, which is randomly generated from a Gaussian distribution of mean zero and variance $\frac{\delta_\tau^2}{\bar{n}}$ such that $\Delta t + \tilde{t}_j > \tilde{t}_{j-1}$ (*i.e.,* adjusting with Δt will not make the timestamp j smaller than the timestamp $(j-1)$). The process is repeated for all signal j ($2 \le j \le n_i$), thus, the time gap between any two consecutive signal j and $(j-1)$ is altered arbitrarily, and because of that, the time length of new gait cycle also fluctuates stochastically.

We summarize in Algorithm 1 the process of generating $p \times m$ timestamps of new gait cycles from the timestamps of m real gait cycles. That is, for each real gait cycle i, we generate p new ones, following the ATD algorithm.

Algorithm 1 Arbitrary Time Deformation

Input

- $\{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_m\}$: timestamps of m real gait cycles;
- p: the number of augmenting cycles per each real cycle;

Output

- $\tilde{\mathcal{T}}$: timestamps of $m \times p$ augmenting cycles;

$\tilde{\mathcal{T}} \leftarrow \{\}$
Compute the time length variance δ_τ^2 by (2)
Compute the averaged number of signals per gait cycle $\bar{n}$ by (4)
for each timestamp vector $\mathbf{t}_i$ **do**
  for l from 1 to p **do**
    $\tilde{\mathbf{t}} \leftarrow \mathbf{t}_i$
    for j from 2 to n_i **do**
      Generate a random value Δt by (6)
      for k from j to n_i **do**
        Adjust $\tilde{t}_k$ with Δt by (5)
      end for
    end for
    $\tilde{\mathcal{T}} \leftarrow \tilde{\mathcal{T}} \cup \tilde{\mathbf{t}}$
  end for
end for
return $\tilde{\mathcal{T}}$

3) THE STOCHASTIC MAGNITUDE PERTURBATION

We present here the Stochastic Magnitude Perturbation (SMP) algorithm which produces augmenting gait cycle by altering the magnitude of each signal in both three axes X, Y and Z. By SMP algorithm, the magnitude is altered randomly, however, the waveform of the based cycle and the smoothness between signals are preserved. Then, to produce the augmenting gait cycle varying in both time and magnitude naturally as real data, we simply perform ATD and SMP sequentially (*i.e.,* the gait cycle augmenting by ATD algorithm is used as the input of SMP). Here, for each user, we use m real cycles and $m \times p$ ATD-based augmenting gait cycles as the input of SMP. Totally, there are $m' = m + p \times m$ inputing gait cycles for each user.

Let $\tilde{\mathbf{A}}$ be the matrix containing magnitude of a gait cycle augmented from the cycle $\mathbf{A}_i$, ($1 \le i \le m'$). In the beginning, $\tilde{\mathbf{A}}$ is initialized from $\mathbf{A}_i$, $\tilde{\mathbf{A}} \leftarrow \mathbf{A}_i$. Then, alteration with random value is performed on $\tilde{\mathbf{A}}$ to make it varying from $\mathbf{A}_i$. To maintain the smoothness and preserve the waveform, instead of modifying each signal separately, we split $\tilde{\mathbf{A}}$ into several *frames* by the signal semantic, then the alteration is performed on each frame. Specifically, in each axis D, ($D \in \{X, Y, Z\}$), the gait cycle is split independently, using the delimitation points $\mathbf{p}_D$ determined by

$$\mathbf{p}_D = \{j \mid 2 \le j \le (n_i - 1), (\tilde{a}_j^D - \bar{a}_D)(\tilde{a}_{j+1}^D - \bar{a}_D) \le 0\}, \tag{7}$$

where $\tilde{a}_j^D$ is the signal j in axis D of cycle $\tilde{\mathbf{A}}$, and $\bar{a}_D$ is the mean of all signals in $\tilde{\mathbf{A}}$ in axis D

$$\bar{a}_D = \frac{1}{n_i} \sum_{j=1}^{n} \tilde{a}_j^D. \tag{8}$$

Equation (7) means that $\mathbf{p}_D$ consists of the positions, at which, the value of signal in axis D is changed from below/above the mean to above/below the mean. Thus, by the points $\mathbf{p}_D$, the gait cycle in each axis D is split into several frames, so that, each frame consists of consecutive signals which are all either above or below the mean.

Let $\mathbf{f}$ denote a frame segmented from $\tilde{\mathbf{A}}$ in an axis D. We identify the signal j^* in $\mathbf{f}$ having the largest absolute magnitude

$$j^* = \underset{j \in [f_s, f_e]}{\operatorname{argmax}} |\tilde{a}_j^D|, \tag{9}$$

where f_s and f_e are the beginning and ending points of $\mathbf{f}$. Next, we estimate the variance δ_{D,j^*}^2 of signal j^* in axis D over m'

input cycles,

$$\delta^2_{D,j^*} = \frac{1}{m'-1}\sum_{i=1}^{m'}(a^D_{i,j^*} - \bar{a}^D_{j^*})^2, \tag{10}$$

where a^D_{i,j^*} is the signal j^* of cycle i in D-axis, and $\bar{a}^D_{j^*}$ is the mean of signal j^* in axis D over m' input cycles

$$\bar{a}^D_{j^*} = \frac{1}{m'}\sum_{i=1}^{m'} a^D_{i,j^*}. \tag{11}$$

Then, a random value Δa is generated according to Gaussian distribution of mean zero and variance δ^2_{D,j^*}, $\Delta a \sim \mathcal{N}(0, \delta^2_{D,j^*})$. For each signal j in $\mathbf{f}$ ($f_s < j \leq f_e$), its value is adjusted by:

$$\tilde{a}^D_j \leftarrow \tilde{a}^D_j + \Delta a \frac{\tilde{a}^D_j - \bar{a}_D}{\tilde{a}^D_{j^*} - \bar{a}_D}. \tag{12}$$

Equation (12) means that in each frame $\mathbf{f}$, a random value Δa is used as the baseline for adjusting all signals in $\mathbf{f}$. The fraction $\frac{\tilde{a}^D_j - \bar{a}_D}{\tilde{a}^D_{j^*} - \bar{a}_D}$ ensures that, the adjusting amount for high-magnitude signal is greater than it for low-magnitude signal. By this fraction, the smoothness of signal in based cycle is preserved in augmenting cycle.

We summarize in Algorithm 2 the process of generating $q \times m'$ new gait cycles from m' gait cycles following the SMP algorithm, *i.e.,* with each gait cycle $\mathbf{A}_i$, q augmenting cycles are produced.

Algorithm 2 Stochastic Magnitude Perturbation

Input

- m' gait cycles $\left\{\mathbf{A}_1, \mathbf{A}_2, \ldots \mathbf{A}_{m'}\right\}$;
- q: the number of augmenting cycles per each real cycle;

Output

- $\tilde{\mathcal{A}}$: a set of $q \times m'$ augmented gait cycles;

$\tilde{\mathcal{A}} \leftarrow \{\}$
for each cycle $\mathbf{A}_i$ **do**
  for k from 1 to q **do**
    $\tilde{\mathbf{A}} \leftarrow \mathbf{A}_i$
    for each axis $D \in \{X, Y, Z\}$ **do**
      Identify the frame delimitation points $\mathbf{p}_D$ by (7)
      Segment $\tilde{\mathbf{A}}$ in D-axis into frames using $\mathbf{p}_D$
      for each frame $\mathbf{f}$ **do**
        Determine the signal j^* having highest absolute magnitude by (9)
        Estimate the variance δ^2_{D,j^*} of signal j^* in axis D by (10)
        Generate a random Δa by $\Delta a \leftarrow \mathcal{N}(0, \delta^2_{D,j^*})$
        for each signal $\tilde{a}^D_j$ in $\mathbf{f}$ **do**
          Adjust value of signal $\tilde{a}^D_j$ by (12).
        end for
      end for
    end for
    $\tilde{\mathcal{A}} \leftarrow \tilde{\mathcal{A}} \cup \tilde{\mathbf{A}}$
  end for
end for
return $\tilde{\mathcal{A}}$

C. CONVOLUTIONAL NEURAL NETWORK

In this section, we present the CNN for gait classification including the shape of input data, the network architecture, and the optimization method.

1) INPUT DATA

With each user in the dataset, both the real and the augmenting gait cycles are used for training the deep network. In total, there are $m \times (1 + p + q + p \times q)$ gait cycles of each user, in which there are m real cycles, $m \times p$ ATD-based augmenting cycles, $m \times q$ cycles augmented by SMP, and $m \times p \times q$ ones augmented by both. First, all gait cycles are normalized to have a fixed number of signals $n = 120$ using spline interpolation [38]. Then, with each signal j in a gait cycle $\mathbf{A}_i$, we determine its magnitude summation $a^M_{i,j} = \sqrt{(a^X_{i,j})^2 + (a^Y_{i,j})^2 + (a^Z_{i,j})^2}$. Each gait cycle $\mathbf{A}_i$ is represented by a matrix of size (4×120) whose rows are formed by the acceleration forces acted on the X, Y, Z, and the magnitude summation, respectively. Finally, each component of $\mathbf{A}_i$ is normalized to zero mean and unit variance over the training dataset, as it leads to faster convergence and higher performance [39].

2) NETWORK ARCHITECTURE

The proposed CNN architecture for gait identification is sketched in Figure 4. The network consists of 4 convolutional layers (Conv) for deep features extraction, and 2 fully connected layers (FC) for classification. The network accepts the gait cycle expressed by a (4×120) matrix $\mathbf{A}_i$ as the input, and outputs a vector

$$\hat{\mathbf{y}}_i = \left[\hat{y}_{i,1}\ \hat{y}_{i,2}\ \ldots\ \hat{y}_{i,N_u}\right], \tag{13}$$

where N_u is the number of users in the dataset, and $\hat{y}_{i,j}$ $(1 \leq j \leq N_u)$ represents the predicted probability of cycle $\mathbf{A}_i$ belongs to the user j. Detailed computation of each layer is explained as follows.

- **Conv1**. Given a gait cycle $\mathbf{A}_i$, the Conv1 performs one-dimensional convolution using 32 filters of size (1×6), and uses the rectified linear unit (ReLU) as the activator, to get the output of size $(32@4 \times 115)$. By the one-dimensional convolution, this layer uses filtering on each dimension of the input data separately, and does not consider the correlation among different dimensions.

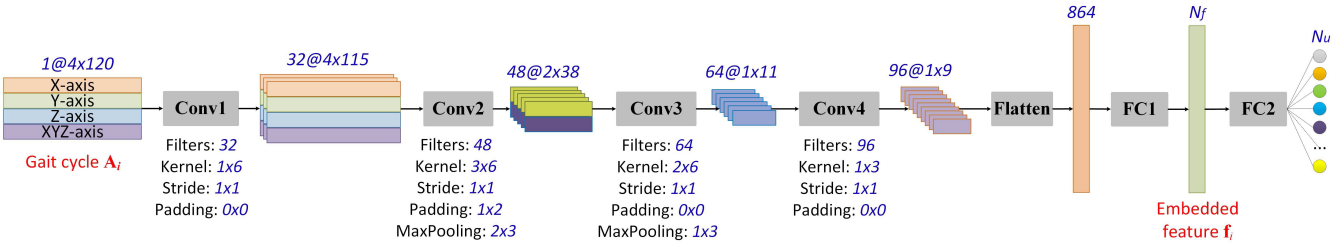


FIGURE 4. The architecture of deep convolutional neural network for gait identification.

- **Conv2**: The second Conv layer performs two-dimensional convolution with 48 filters of size (3×6), activated by the ReLU unit. By these filters, the correlation information between data in different dimensions is detected and captured automatically. Then, the max-pooling of size (2×3) is used to obtain the output of size $(48@2 \times 38)$.
- **Conv3**: The third Conv layer creates a fusion of the data in two dimensions to get a one-dimensional vector using 64 filters of size (2×6), activated by the ReLU unit. Then, the max-pooling of size (1×3) is used to reduce the data size to $(64@1 \times 11)$.
- **Conv4**: The last Conv layer performs convolution with 96 filters of size (1×3) as the final filtering to get an output of size $(96@1 \times 9)$. The ReLU unit is also used as the activator. Then, the output is flattened into a vector of size 864, which is represented by 864 neuron nodes.
- **FC1**: The first FC layer computes N_f outputs from 864 inputs, in which, each output is a linear combination of all the input nodes, and activated by the Hyperbolic tangent function. All the output nodes form a vector

$$\mathbf{f}_i = \left[f_{i,1}\ f_{i,2}\ \ldots\ f_{i,N_f}\right] \tag{14}$$

  whose elements are used as the hidden features of the gait cycle $\mathbf{A}_i$ extracted by CNN.
- **FC2**: The second FC layer acts as a classifier, which uses $\mathbf{f}_i$ to predict the user identity. For a model of N_u users, this layer outputs a vector $\hat{\mathbf{y}}_i$ of size N_u, where each component $\hat{y}_{i,j}$ is computed from all the input nodes, activated by the log softmax function.

In this network, N_f is determined according to the number of participated users N_u as $N_f = \frac{3}{2}N_u$. In overall, a better performance can be achieved by increasing N_f, however, too large N_f could cause the curse of dimensionality. In this paper, we select an optimal value for N_f. Detailed analysis of N_f on the classification performance is presented in [10].

3) NETWORK OPTIMIZATION

The network parameters are initialized randomly, then updated iteratively through the supervised training process using the stochastic gradient descent algorithm. Specifically, let $\mathcal{L}$ be the set of gait cycles for training the model. Each gait cycle $\mathbf{A}_i \in \mathcal{L}$ is labeled with the identity of its owner user u_i $(1 \le u_i \le N_u)$. The training process is repeated by a number of epochs. In each epoch, a set $\mathcal{B} \subset \mathcal{L}$ including B gait cycles are selected, where B is the batch size. All gait cycles in $\mathcal{B}$ are input to the network to obtain a set of B output vectors $\hat{\mathbf{y}}_i$ (13), $(1 \le i \le B)$. The loss over the batch $\mathcal{B}$ is computed using the negative log-likelihood (NLL) function

$$L = -\frac{1}{B}\sum_{i=1}^{B}\log(\hat{y}_{i,u_i}). \tag{15}$$

The loss L is used to update the network parameters through the back-propagation procedure. The process is repeated to minimize L iteratively by rotating the gait cycles in $\mathcal{B}$ until it spans entire $\mathcal{L}$. When all gait cycles in $\mathcal{L}$ have been used, the training process for one epoch completes. Then, entire process is repeated with a new epoch until the stopping criterion is met (see Section IV-A).

IV. EVALUATION

In this section, we present the experiment and evaluating result of the proposed model. First, we describe the datasets and experimental procedure. Then, the results of identification and verification tasks are reported along with the detailed analysis in each task. Finally, we give a relative comparison between our study and state-of-the-art researches.

A. DATASET AND EXPERIMENT PROCEDURE

We evaluated our model using 2 datasets, the realistic and the largest one, respectively published by Chonnam National University and Osaka University.

1) CHONNAM NATIONAL UNIVERSITY DATASET

The Chonnam National University dataset (CNU) [20] comprises of acceleration and orientation gait signals of 38 volunteers (10 females and 28 males), acquired by Google Nexus One. Each volunteer conducted 19 walking sessions on average, each session was performed on a different day during a long-time period, and under unconstrained outside conditions (*e.g.,* clothes, foot gears, road surfaces). Thus, this dataset is considered as the most realistic one. By it, we aim to confirm our methods on real conditions.

Using the preprocessing methods described in Section III-A, we processed the raw gait signals and segmented to gait cycles. There were 21427 extracted gait cycles

of all users. The extracted gait cycles were divided into 3 parts. The first one, denoted as $\mathcal{L}$, was formed by randomly selecting m gait cycles of each user, and was used for training the model. From the remaining data, 50% gait cycles of each user were selected randomly to make the validation dataset $\mathcal{V}$. All other gait cycles formed the testing dataset $\mathcal{T}$.

In the training phase, gait cycles in $\mathcal{L}$ were augmented to generate more training data using the proposed augmentation algorithms (section III-B). The set $\mathcal{V}$ were used to terminate the training process. Specifically, when one epoch was completed, the loss on validation set $\mathcal{V}$ was computed. If the loss on $\mathcal{V}$ did not decrease during 15 consecutive epochs, the training process was terminated and all parameters of the network were fixed for testing. In the testing phase, each gait cycle in $\mathcal{T}$ was input to the trained model to get the predicted user identity. The identification accuracy is computed by

$$ACC = \frac{100 \times N_c}{|\mathcal{T}|}, \tag{16}$$

where N_c is the correct classifying cases, and $|\mathcal{T}|$ is the number of gait cycles in the testing set $\mathcal{T}$.

To provide a detailed analysis on the impact of the augmentation algorithms, we conducted the above experiment with different numbers of augmenting gait cycles (*i.e.*, p, q), performed on different amount of training gait cycles (*i.e.*, m). Each case was repeated 10 times and the averaged accuracy was reported.

2) OSAKA UNIVERSITY GAIT DATASET

The Osaka University inertial sensor-based gait dataset (OU-ISIR) [22] is considered as the largest published inertial sensor-based gait dataset, which consists of two sub-datasets. The first one contains gait signals of 744 users (each user performed two level-walk sequences), captured by the acceleration and gyroscope sensors of an IMU unit placing in the center of the back waist. The second sub-dataset comprises data of 495 subjects, each one walked 4 sequences including 2 level-walk, one up-slope, and one down-slope, collected using 3 IMU units placing in the left, right and center of the back waist, respectively. By this dataset, we aim to analyze our model on a large number of users, thus, the first sub-dataset was used to maximize the number of users.

Using the method summarized in Section III-A, we also preprocessed and extracted the gait cycles from the raw gait signals. However, for this dataset, the disorientation elimination step was skipped as the coordination of devices was fixed during the time of acquiring data [22]. Totally, there were 13023 gait cycles extracted. 8 gait cycles of each user were used to construct the training dataset $\mathcal{L}$; 50% of the remaining gait cycles formed the validation dataset $\mathcal{V}$, and all the rest gait cycles were used for testing. ATD and SMP algorithms were also adopted on $\mathcal{L}$ to get more training data. However, this dataset does not contain the timestamp of gait signal, which is needed for the ATD algorithm. To overcome this issue, with each gait cycle i, we approximately assigned the timestamp of each signal as $t_{i,j} = 10 \times (j-1)$ ms, $1 \leq j \leq n_i$. This means, in each gait cycle, the timestamp of first signal is $t_{i,0} = 0$, and the timestamp of each signal after that is gradually increased by 10 ms. The reason for this approximation is the device sampled signals at the frequency of 100 Hz when collecting this dataset. Thus, the averaged time gap between two consecutive signals is 10 ms.

For this dataset, different settings of p and q were also evaluated. However, as each user just performed two short walking sequences, the number of gait cycles per each user in this dataset is limited, which is 17 on average. Thus, all experiments were conducted with $m = 8$.

B. VISUALIZATION OF THE AUGMENTING GAIT CYCLE

First, we provide a visualization of the augmenting gait cycles to confirm that the proposed algorithms could generate new gait cycles with the desired properties. Figures 5 and 6 plot 2 gait cycles randomly selected in CNU and OU-ISIR datasets along with their augmenting cycles, respectively.

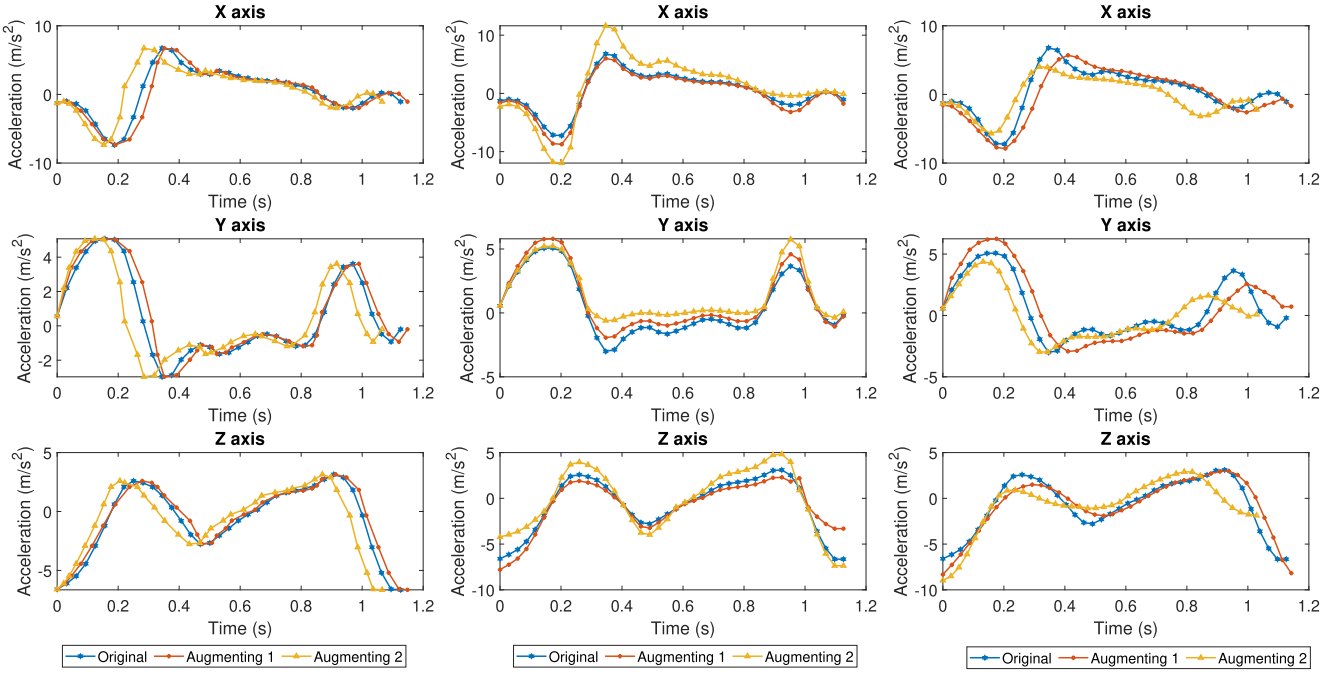
We could see that, when using the ATD algorithm (Figures 5a, 6a), the magnitude of gait signal is preserved in the augmenting cycles. And, the acting time of each signal fluctuates arbitrarily. On the other hand, for the gait cycles generated from the SMP algorithm (Figures 5a, 6a), the acting time of each signal is preserved, and the magnitude varies randomly in the augmenting cycles. Finally, by combining the impact of ATD and SMP, the generated gait cycles vary in both magnitude and time (Figures 5c, 6c), and the augmenting gait cycles vary more naturally as had been acquired in real condition. In all cases, the augmenting gait cycles still maintain the waveform of their based cycle, and they still represent a same user identity in the vision manner.

C. IDENTIFICATION PERFORMANCE

1) CNU DATASET

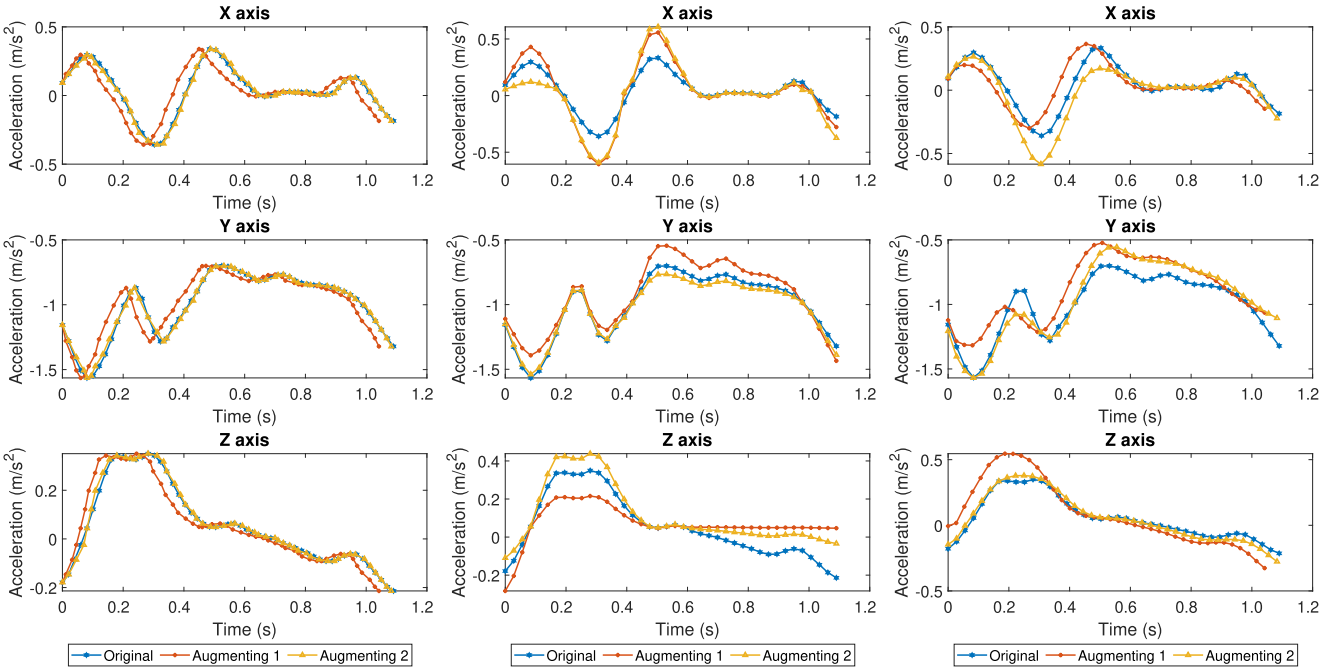
The effectiveness of our proposed augmentation algorithms on the CNU dataset is illustrated in Figure 7, which provides the identification performance under different amounts of augmenting and training data.

First, we provide in Figures 7a and 7b the accuracy comparison between adopting ATD or SMP alone and no data augmentation, under different amount of real training data (*i.e.,* from 10 to 60 gait cycles). As we can see, ATD and SMP clearly improve the identification accuracy, and their efficiency depends on the number of training gait cycles m and the augmenting gait cycles p, q. When having little training data, the amount of increasing accuracy by applying data augmentation is higher than that of having much training data. For instance, with $m = 10$ and no data augmentation, the model achieved the ACC of 80.39 %. By using ATD augmentation with $p = 1$, the ACC was increased to 83.01 %, thus the gained ACC for this case was 2.61 %. When m was increased to 60, the gained ACC gradually decreased to 0.78 %. Similarly, when adopting SMP with $p = 1$ and $m = 10$, the gained ACC was 0.99 %, which was reduced to 0.85% when m was increased to 60.



(a) ATD Algorithm (b) SMP Algorithm (c) ATD and SMP combination

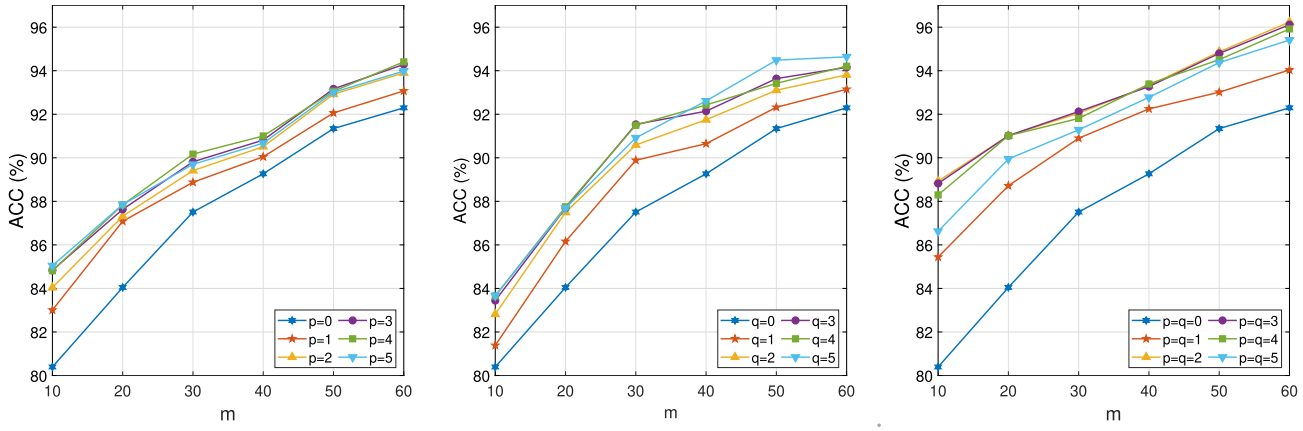
FIGURE 5. The augmenting gait cycles in time, magnitude and both, of a randomly selected cycle in CNU dataset, generated from ATD, SMP and their combination, respectively.



(a) ATD Algorithm (b) SMP Algorithm (c) ATD and SMP combination

FIGURE 6. The augmenting gait cycles in time, magnitude and both, of a randomly selected cycle in OU-ISIR dataset, generated from ATD, SMP and their combination, respectively.

Besides, the effectiveness of augmenting algorithms is impacted by the amount of augmenting cycles. For both ATD and SMP, increasing the number of augmenting gait cycles to 2 and 3 could further improve the accuracy. For instance, with $m = 10$, when using ATD algorithm with $p = 2$ and $p = 3$, the accuracy was 84.04% and 84.84 %, respectively. In a same way, by increasing q to 2 and 3 when adopting SMP algorithm, the ACC was raised to 82.82% and 83.44 %, respectively. A similar impact could be observed with other values of m. However, further increasing the number of



(a) ATD (b) SMP (c) ATD and SMP

FIGURE 7. The identification ACC (%) measured by CNU dataset, under different settings of augmenting gait cycles, where $p = 0$, $q = 0$, and $p = q = 0$ mean no data augmentation.

augmenting cycles beyond 3 just resulted in minor improving the ACC, or sometimes degrading. The reason for this fact is, with a large number of augmenting cycles, the redundant data could exist and causes inefficient learning. Thus, we suggest that the appropriate number of augmenting gait cycles should be 2 or 3 when adopting ATD or SMP separately.

With the combination of ATD and SMP, the augmenting gait cycle varies from each other more naturally (see Section IV-B). So, as we expected, it resulted in much higher accuracy comparing to using each algorithm separately, as displayed in Figure 7c. For instance, when using $m = 10$ training gait cycles, combining ATD and SMP with $p = q = 1$ and $p = q = 2$ could increase the ACC from 80.39% to 85.44% and 88.93 %, respectively. When $m = 60$, the ACC was boosted from 92.3% to 94.04% and 96.24% for the case of $p = q = 1$ and $p = q = 2$, respectively. Similar to using ATD or SMP alone, over-increasing p and q causes generating redundant augmenting gait cycles and degrading the ACC. In addition, by combining the ATD and SMP algorithms, the number of augmenting gait cycles is increased multiplicatively when increasing p and q. For instance, with $p = q = 1$, there are 3 augmenting gait cycles for each real cycle. And, with $p = q = 2$ and $p = q = 3$, the number of augmenting gait cycles respectively are 8 and 15 for each real cycle. So, when using ATD and SMP at the same time, we recommend using $p = q = 1$ or $p = q = 2$ to avoid generating redundant augmenting gait cycles.

Note that, when combining ATD and SMP, p and q can be different (*e.g.,* $p = 1$ and $q = 3$; $p = 3$ and $q = 1$). However, such settings will lead to a large number of time-based or magnitude-based augmenting gait cycles. Then, the ACC will be degraded as we just showed above. Thus, we recommend that p and q should be equal to get optimal performance.

2) OU-ISIR DATASET

Table 1 summarizes the identification performance measured on OU-ISIR dataset when training with 8 gait cycles per each user, under different settings of data augmentation.

TABLE 1. The identification ACC (%) when adopting different settings of data augmentation on 8 real gait cycles per each user, measured on OU-ISIR dataset.

Augmentation algorithm	*Number of augmenting gait cycles*				
	1	*2*	*3*	*4*	*5*
ATD	84.91	87.22	87.30	87.22	86.98
SMP	84.71	87.44	87.41	87.26	86.98
ATD & SMP	87.84	89.99	89.78	89.42	88.25
None	82.53				

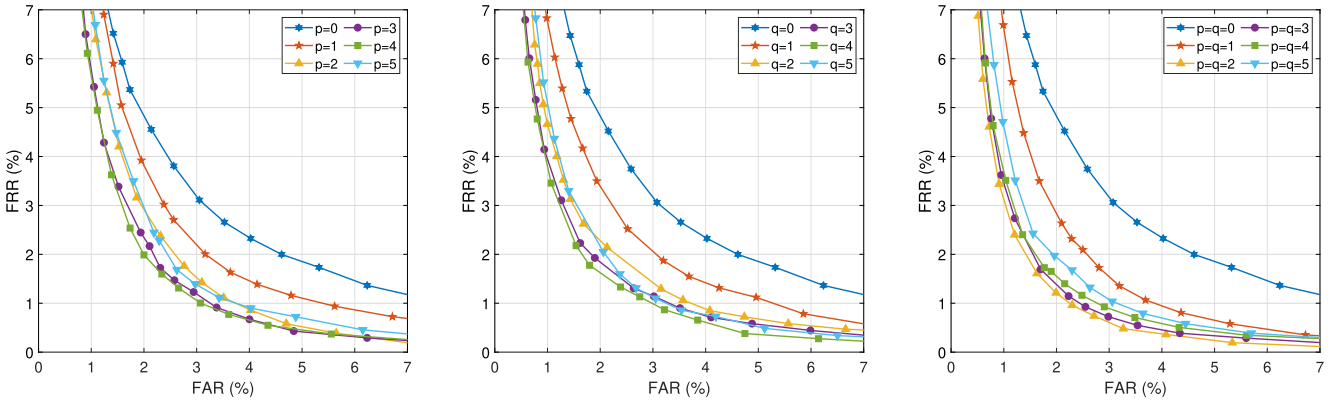
We could see that the proposed augmentation algorithms are also effective for dataset having a large number of subject as OU-ISIR. When there was no data augmentation, the identification ACC was 82.53 %. By using ATD with $p = 1$ and $p = 2$, the ACC was improved to 84.91% and 87.22 %, respectively. Similarly, adopting SMP alone could boost the performance to 84.71% and 87.44% for $q = 1$ and $q = 2$, respectively. The best ACC experimented in OU-ISIR, was 89.99 %, achieved by combining ATD and SMP with $p = q = 2$. For this dataset, we observed a same inference as it was with CNU dataset. That is, too much augmenting data could lead to redundant training cycles and make the performance degrading. The number of augmenting gait cycles per each real cycle (*i.e.,* p, q) should not exceed 2.

D. VERIFICATION PERFORMANCE

In this section, we extend the identification model (Figure 4) to construct a verification model using transfer learning and one-class support vector machine (OCSVM) [31]. The performance evaluating on CNU and OU-ISIR datasets is reported to confirm the effectiveness on verification task.

1) OCSVM-BASED VERIFICATION MODEL

It is well-known that, a deep network, which has been trained for a specific task (*e.g.,* identification), can be transferred to



(a) ATD (b) SMP (c) ATD and SMP

FIGURE 8. The ROC curves of the verification performance under different data augmentation settings, using 60 training gait cycles, measured by CNU dataset.

use in a related task (*e.g.,* verification) [40]. A trained CNN network can be used as a universal feature extraction tool, which automatically extracts good features, even for subjects that have not leaned before [41].

Following that approach, we construct a verification model as follows. First, we train the CNN network (Figure 4) with the perspective of identification as described in Section III-C.3. At this step, the ATD and SMP algorithms are also adopted to generate more training data. Upon the training is completed, we discard the FC2 layer, and use the remaining network as a generic feature extraction tool which outputs a feature vector $\mathbf{f}_i$ (14) when inputing a gait cycle $\mathbf{A}_i$. Then, the extracted vectors $\mathbf{f}_i$ are used as the input of OCSVM to build a verification model.

For the verification model of a user u, OCSVM uses a set of feature vectors $\mathbf{f}_i$ belong to u to learn a *boundary* which covers all $\mathbf{f}_i$ and separates them from those of other users. By OCSVM, a classifier is trained using only the data of positive class (*i.e.,* user u), but can handle the impostors whose data have not given in the training stage [31]. The main strategy of OCSVM is to map the input data to an appropriate space corresponding to the kernel, then build a hyperplane to separate them from the origin with maximum margin.

At the run time stage, a testing gait cycle $\mathbf{A}'$ is input to the network to get a feature vector $\mathbf{f}'$. The learned OCSVM model receives $\mathbf{f}'$ and returns a score $\mathsf{g}(\mathbf{f}')$ which reflects the position of $\mathbf{f}'$ in relation to the boundary. A positive $\mathsf{g}(\mathbf{f}')$ means that $\mathbf{f}'$ is inside the boundary, zero indicates $\mathbf{f}'$ lies on the boundary, and outside otherwise. Thus, when $\mathsf{g}(\mathbf{f}') \geq 0$, $\mathbf{A}'$ is classified as positive class, and negative if $\mathsf{g}(\mathbf{f}') < 0$. We use false acceptance rate (FAR), false rejection rate (FRR) and equal error rate (EER) to measure the verification performance. The FRR is computed as the percentage of positive cases but are classified as negative, while the FAR is determined by the percentage of negative cases but the model results to positive. To provide a trade-off between FAR and FRR, we adopt an additional parameter θ, which plays as a flexible threshold to determine positive or negative. Specifically, given a specific θ, $\mathbf{A}'$ is classified as positive if $\mathsf{g}(\mathbf{f}') \geq \theta$, and negative if $\mathsf{g}(\mathbf{f}') < \theta$. The equal error rate (EER) is determined as the average of FAR and FRR corresponding to a specific threshold θ, at which, FAR and FRR are equivalent or approximate to each other.

TABLE 2. The verification EER (%) when using 60 gait cycles for training, under different numbers of augmenting gait cycles, measured on CNU dataset.

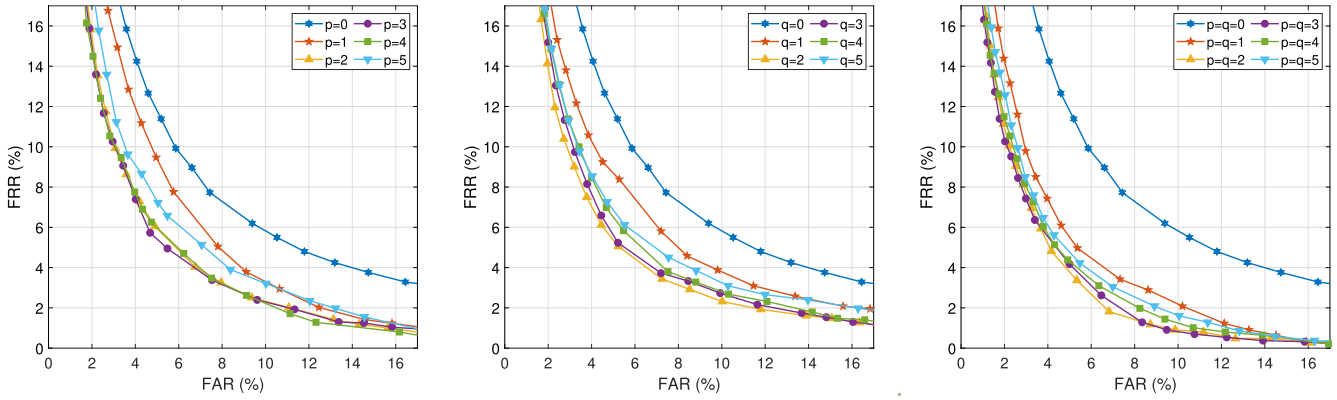
Augmentation Algorithm	*Number of augmenting gait cycles*				
	1	*2*	*3*	*4*	*5*
ATD	2.63	2.35	2.15	1.99	2.29
SMP	2.52	2.14	1.91	1.79	2.06
ATD & SMP	2.30	1.62	1.70	1.75	1.97
None	3.07				

2) CNU DATASET

Figure 8 displays the verification performance ROC curves measured in CNU dataset, under different settings of data augmentation when using 60 real gait cycles for training, and Table 2 provides the EER for each case. Specifically, the EER when using no data augmentation was 3.07 %. By adopting ATD alone with $p = 1$, $p = 2$ and $p = 3$, the EER was reduced to 2.63 %, 2.35% and 2.15 %, respectively. For SMP, a similar effect could be observed. Using SMP along with $q = 1$, $q = 2$, and $q = 3$ could reduce the EER to 2.52 %, 2.14% and 1.91 %, respectively. After all, as expected, the highest performance (lowest EER) was achieved as 1.62% by combining ATD and SMP with $p = q = 2$.

3) OU-ISIR DATASET

Figure 9 displays the ROC curve of FAR and FRR measured on OU-ISIR dataset when using 8 real gait cycles of each user for training, under different configurations of data augmentation. Table 3 summarizes those results by providing the EER



(a) ATD (b) SMP (c) ATD and SMP

FIGURE 9. The verification EER (%) when using 8 gait cycles for training, under different numbers of augmenting gait cycles, measured on OU-ISIR dataset.

TABLE 3. The verification EER (%) measured on OU-ISIR dataset, when training with 8 gait cycles, and using different numbers of augmenting gait cycles.

Augmentation	*Number of augmenting gait cycles*				
Algorithm	*1*	*2*	*3*	*4*	*5*
ATD	6.76	5.48	5.21	5.49	6.02
SMP	6.5	5.14	5.23	5.66	5.82
ATD & SMP	5.17	4.49	4.59	4.65	4.85
None	7.58				

for each case. By these results, we show that the proposed data augmentation algorithms are also effective for the task of user verification, under a large number of subjects. Specifically, without data augmentation, the model achieved the EER of 7.58 %. The use of ATD algorithm with $p = 3$ could reduce it to 5.21 %. Similarly, by using SMP with $q = 2$, the EER was decreased to 5.14 %. Again, the best performance experimented in this dataset was 4.49 %, achieved by combining ATD and SMP with $p = q = 2$.

E. STATE-OF-THE-ART RESEARCH COMPARISON

In this section, we provide a comparison between our study and state-of-the-art researches which experimented on CNU and OU-ISIR datasets. It is very difficult to provide a fair comparison as the experimental evaluation of each research was conducted in different settings (*e.g.,* the portion of training data, the potion of data for each testing trial, the number of users). As the classification performances (*e.g.,* FAR, FRR, EER) are strongly affected by these factors [7], the comparison is just relative. We summarize in Table 4 not only classification performances but also other settings, to provide a comparison as fairly as possible, where the cell with '−' means the corresponding factor was not given.

On the CNU dataset, our study clearly out-performs the state-of-the-art research [20] in all aspects. Specifically, in their research, they used 50% of data for training, and 4 gait cycles for each time of testing. They achieved 94.93% ACC for identification task and 5.35% EER for verification task. By data augmentation, we could use smaller potion of data for training as 23% dataset. However, our trained model is more robust as it achieves higher performance (*i.e.,* 96.24% ACC and 1.62% EER) while using shorter data sequences for each testing trial (*i.e.,* 1 gait cycle).

The OU-ISIR dataset has been used for experimental evaluation on many researches. In the early time, Zhong *et al.,* 2014 [42] constructed the gait dynamic images (GDIs), which is invariant to sensor orientation, for the task of user verification. The EER of 5.6% was reported when using 50% of data for training and one walking session (9m long) for each time of testing. The studies after that gradually improved the classification performance overtime [6], [14]–[16], [36], [43]. A recent research by Delgado *et al.* [14] reported a high-performance model in both the task of verification and identification (*i.e.,* 94.8 % ACC and 1.1% EER) with the use of CNN on multi-sensor and multi-task learning. On the other hand, Fernandez *et al.* [16] conducted a first approach with RNN and achieved the EER of 7.55% when using 80% of data for training and 1 cycles for each testing trial. To the best of our knowledge, [16] is also the only research that adopts RNN to the task of inertial sensor-based gait recognition. Usually, the classification performance is greatly impacted by the potion of training data and the amount of data used for each testing trial. A larger potion data for training usually results in a more robust classifier, while a longer sequence of data for testing provides more stable and sufficient information to get higher prediction accuracy.

We can see that our model out-performs all other methods while using smaller potion of training and each testing trial, excepting the studies [14], [36]. The reported ACC in [36] is 96.84 %, which is higher than our identification performance as 89.99 %. However, in [36], the authors just used 100 users selected randomly among 744 users for experiment, while

TABLE 4. The comparison on different factors (*e.g.,* dataset, performance, training/testing data potion) between state-of-the-art gait recognition researches.

Dataset	*Study*	*Training potion*	*Testing potion*	*Identification* (ACC)	*Verification* (EER)
CNU [20]	Hoang *et al.*, 2015 [20]	50 % dataset	4 cycles	94.93 %	5.35 %
	Our study	**60 cycles/user (≈ 23 % dataset)**	**1 cycle**	**96.24 %**	**1.62 %**
OU-ISIR [22]	Zhong *et al.*, 2014 [42]	50 % dataset	9m long	–	5.6 %
	Sprager *et al.*, 2015 [6]	–	1.4s	–	10.07 %
	Wei *et al.*, 2015 [43]	50 % dataset	≈ 3m long	83.8 %	–
	[(1)]Nguyen *et al.*, 2017 [36]	75 % samples	100 signals	96.84 %	10.43 %
	Subramanian *et al.*, 2018 [15]	50 % dataset	1 cycle	–	> 6 %
	Delgado *et al.*, 2018 [14]	50 % dataset	9m long	94.8 %	1.1 %
	Fernandez *et al.*, 2019 [16]	80 % dataset	1 cycle	–	7.55 %
	[(2)]**Our Study**	**8 cycles/user (≈46 % dataset)**	**1 cycle**	**89.99 %**	**4.49 %**

[(1)]Identification ACC was experimented on 100 users selected randomly; [(2)]only using accelerometer.

it has been showed that, larger participated subject would reduce the classification performance [7], [22]. Similarly, in the study [14], beside the robust of their proposed method, the high performance has the contribution of long testing data sequence (*i.e.,* 9m long) and the fusion of multiple sensors (*i.e.,* gyroscope and accelerometer). For this study, we focus on designing a gait data-based augmentation algorithms. Thus, we only use the accelerometer. In addition, we used a short sequence data as 1 gait cycle for each testing trial. We strongly believe that, incorporating the gyroscope data and increasing the training/testing data amount could increase the performance as indicating in prior studies [7], [16].

V. CONCLUSION

In this study, we proposed a novel approach for inertial sensor-based gait data augmentation. First, two algorithms as ATD and SMP were proposed, to create random variation in the time and magnitude domain, respectively. They were designed to arbitrarily adjust the real gait data for making variation, while preserving the critical information of real data in augmenting data. A generic deep CNN gait model was designed for user identification task, in which, the ATD and SMP algorithms were coordinated effectively. Then, a verification model was constructed by extending the identification model using transfer learning and OCSVM. We analyzed ATD, SMP and the proposed models using the realistic and the largest gait dataset (*i.e.,* CNU dataset and OU-ISIR dataset, respectively). The experiment results show that the combination of ATD and SMP could generate augmenting data varying naturally as real data. And by their help, the verification and identification performances were effectively improved. In addition, the proposed algorithms could operate on various sensor models, which could be widely employed for data augmentation in future deep gait models.

REFERENCES

[1] L. Lee and W. Grimson, "Gait analysis for recognition and classification," in *Proc. 5th IEEE Int. Conf. Autom. Face Gesture Recognit.*, Jun. 2003, pp. 155–162.

[2] D. Tao, X. Li, X. Wu, and S. J. Maybank, "General tensor discriminant analysis and Gabor features for gait recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1700–1715, Oct. 2007.

[3] L. Middleton, A. Buss, A. Bazin, and M. Nixon, "A floor sensor system for gait recognition," in *Proc. 4th IEEE Workshop Autom. Identificat. Adv. Technol. (AutoID)*, Oct. 2006, pp. 171–176.

[4] T. C. Pataky, T. Mu, K. Bosch, D. Rosenbaum, and J. Y. Goulermas, "Gait recognition: Highly unique dynamic plantar pressure patterns among 104 individuals," *J. Roy. Soc. Interface*, vol. 9, no. 69, pp. 790–800, Apr. 2012.

[5] H. J. Ailisto, M. Lindholm, J. Mantyjarvi, E. Vildjiounaite, and S.-M. Makela, "Identifying people from gait pattern with accelerometers," *Proc. SPIE*, vol. 5779, pp. 7–14, Mar. 2005.

[6] S. Sprager and M. B. Juric, "An efficient HOS-based gait authentication of accelerometer data," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 7, pp. 1486–1498, Jul. 2015.

[7] S. Sprager and M. Juric, "Inertial sensor-based gait recognition: A review," *Sensors*, vol. 15, no. 9, pp. 22089–22127, Sep. 2015.

[8] L. Tran, T. Hoang, T. Nguyen, and D. Choi, "Improving gait cryptosystem security using Gray code quantization and linear discriminant analysis," in *Proc. Int. Conf. Inf. Secur.* Cham, Switzerland: Springer, 2017, pp. 214–229.

[9] G. Giorgi, F. Martinelli, A. Saracino, and M. Sheikhalishahi, "Try walking in my shoes, if you can: Accurate gait recognition through deep learning," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.* Cham, Switzerland: Springer, 2017, pp. 384–395.

[10] M. Gadaleta and M. Rossi, "IDNet: Smartphone-based gait recognition with convolutional neural networks," *Pattern Recognit.*, vol. 74, pp. 25–37, Feb. 2018.

[11] O. Dehzangi, M. Taherisadr, and R. Changalvala, "IMU-based gait recognition using convolutional neural networks and multi-sensor fusion," *Sensors*, vol. 17, no. 12, p. 2735, Nov. 2017.

[12] Y. Zhao and S. Zhou, "Wearable device-based gait recognition using angle embedded gait dynamic images and a convolutional neural network," *Sensors*, vol. 17, no. 3, p. 478, Feb. 2017.
[13] C. Wan, L. Wang, and V. V. Phoha, "A survey on gait recognition," *ACM Comput. Surv.*, vol. 51, no. 5, p. 89, 2019.
[14] R. Delgado-Escano, F. M. Castro, J. R. Cozar, M. J. Marin-Jimenez, and N. Guil, "An end-to-end multi-task and fusion CNN for inertial-based gait recognition," *IEEE Access*, vol. 7, pp. 1897–1908, 2019.
[15] R. Subramanian and S. Sarkar, "Evaluation of algorithms for orientation invariant inertial gait matching," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 2, pp. 304–318, Feb. 2019.
[16] P. Fernandez-Lopez, J. Liu-Jimenez, K. Kiyokawa, Y. Wu, and R. Sanchez-Reillo, "Recurrent neural network for inertial gait user recognition in smartphones," *Sensors*, vol. 19, no. 18, p. 4054, Sep. 2019.
[17] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intell. Syst.*, vol. 24, no. 2, pp. 8–12, Mar. 2009.
[18] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1717–1724.
[19] J. Frank, S. Mannor, and D. Precup. (2010). *Data Sets: Mobile Phone Gait Recognition Data*. [Online]. Available: http://www.cs.mcgill.ca/~jfrank8/data/gait-dataset.html
[20] T. Hoang, D. Choi, and T. Nguyen, "On the instability of sensor orientation in gait verification on mobile phone," in *Proc. 12th Int. Conf. Secur. Cryptogr.*, vol. 4, 2015, pp. 148–159.
[21] M. Muaaz and C. Nickel, "Influence of different walking speeds and surfaces on accelerometer-based biometric gait recognition," in *Proc. 35th Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2012, pp. 508–512.
[22] 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, Jan. 2014.
[23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[24] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, p. 60, 2019.
[25] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.
[26] J. Wang and L. Perez, "The effectiveness of data augmentation in image classification using deep learning," *Convolutional Neural Netw. Vis. Recognit.*, Dec. 2017.
[27] C. N. Vasconcelos and B. N. Vasconcelos, "Convolutional neural network committees for melanoma classification with classical and expert knowledge based image transforms data augmentation," vol. 1, 2017, *arXiv:1702.07025*. [Online]. Available: https://arxiv.org/abs/1702.07025
[28] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (VTLP) improves speech recognition," in *Proc. Workshop Deep Learn. Audio, Speech Lang. (ICML)*, vol. 117, 2013.
[29] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 23, no. 9, pp. 1469–1477, Sep. 2015.
[30] Z. Tüske, P. Golik, D. Nolden, R. Schlüter, and H. Ney, "Data augmentation, feature combination, and multilingual neural networks to improve ASR and KWS performance for low-resource languages," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc.*, 2014.
[31] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 582–588.
[32] M. P. Murray, A. B. Drought, and R. C. Kory, "Walking patterns of normal men," *J. Bone Joint Surg.*, vol. 46, no. 2, pp. 335–360, 1964.
[33] M. P. Murray, "Gait as a total pattern of movement: Including a bibliography on gait," *Amer. J. Phys. Med. Rehabil.*, vol. 46, pp. 290–333, Feb. 1967.
[34] O. Incel, "Analysis of movement, orientation and rotation-based sensing for phone placement recognition," *Sensors*, vol. 15, no. 10, pp. 25474–25506, Oct. 2015.
[35] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
[36] K.-T. Nguyen, T.-L. Vo-Tran, D.-T. Dinh, and M.-T. Tran, "Gait recognition with multi-region size convolutional neural network for authentication with wearable sensors," in *Proc. Int. Conf. Future Data Secur. Eng.* Cham, Switzerland: Springer, 2017, pp. 197–212.
[37] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, no. 7, pp. 909–996, 1988.
[38] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM J. Numer. Anal.*, vol. 17, no. 2, pp. 238–246, Apr. 1980.
[39] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade*. Berlin, Germany: Springer, 2012, pp. 9–48.
[40] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 806–813.
[41] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.
[42] Y. Zhong and Y. Deng, "Sensor orientation invariant mobile gait biometrics," in *Proc. IEEE Int. Joint Conf. Biometrics*, 2014, pp. 1–8.
[43] Z. Wei, W. Qinghui, D. Muqing, and L. Yiqi, "A new inertial sensor-based gait recognition method via deterministic learning," in *Proc. 34th Chin. Control Conf. (CCC)*, Jul. 2015, pp. 3908–3913.

LAM TRAN received the B.S. degree in computer science from the Ho Chi Minh University of Science, Ho Chi Minh, Vietnam, in September, 2012, and the M.E. degree in computer science from Chonnam National University, Gwangju, South Korea, in August, 2017. He is currently pursuing the Ph.D. degree with the Electronics and Computer Department, Chonnam National University, South Korea. He is also a Lecturer with the Information Technology Faculty, Ho Chi Minh University of Science. His research interests currently focus on inertial sensor-based gait biometrics, privacy enhancing for biometric systems, and mobile security.

DEOKJAI CHOI received the B.S. degree from the Department of Computer Engineering, Seoul National University, in 1982, the M.S. degree from the Department of Computer Science, KAIST, South Korea, in 1984, and the Ph.D. degree from the Department of Computer Science and Telecommunications, University of Missouri-Kansas City, USA, in 1995. He is currently a Full Professor with the School of Electronics and Computer Engineering, Chonnam National University, South Korea. His interests on research are mobile security, human activity profile, biometric authentication, and network security.