

Research Article

Heel-Strike and Toe-Off Detection Algorithm Based on Deep Neural Networks Using Shank-Worn Inertial Sensors for Clinical Purpose

Dmitry Skvortsov ¹, Denis Chindilov,² Nikita Painev,² and Alexey Rozov²

¹Pirogov Russian National Research Medical University, Moscow 101000, Russia

²Neurosoft LLC, Ivanovo 153000, Russia

Correspondence should be addressed to Dmitry Skvortsov; skvortsov.biom@gmail.com

Received 8 October 2022; Revised 31 March 2023; Accepted 17 April 2023; Published 2 May 2023

Academic Editor: Aijun Yin

Copyright © 2023 Dmitry Skvortsov et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A foot placement of inertial sensors is commonly used for heel-strike (HS) and toe-off (TO) event detection. However, in clinical practice, such sensor placement may be difficult or even impossible due to the deformity of patients' feet. The first contribution of this paper is a new algorithm for HS and TO event detection for cases when the sensors are placed on the lateral malleolus. Such sensor placement allows gait analysis in patients with foot deformities. In addition, the placement of the sensor directly on the wide bone surface of the lateral malleolus ensures secure fixation of the sensor during walking. The proposed algorithm is based on deep neural networks, which can be easily adapted (by retraining the neural networks) for analysis of various pathological gait patterns. It is especially important in clinical practice when the number of possible pathological gait patterns is very large. The algorithm proposed in this paper was implemented in a new wearable system for the clinical gait analysis. The second contribution is a validation of this new wearable system. The performance of both proposed algorithm and gait analysis system was evaluated against a reference treadmill system where a capacitance-based pressure platform was used. A total of 117 healthy volunteers participated in the comparison (62 males and 55 females, age 24–55 years, height 162–183 cm). They were asked to perform 2 min walking trials with different speed. Mean accuracy \pm precision was -0.021 ± 0.091 s for gait cycle, 0.589 ± 1.144 steps/min for cadence, $-0.051 \pm 0.544\%$ for stance phase, $-0.37 \pm 0.649\%$ for single support, $0.296 \pm 0.711\%$ for double support, $0.132 \pm 0.561\%$ for load response, and $0.106 \pm 0.661\%$ for preswing. Limitations of the proposed algorithm and its comparison with state-of-the-art algorithms were discussed.

1. Introduction

Gait analysis is one of the most widely applied methods for diagnostics of movement disorders [1–4]. Today, there are gait analysis systems using different technologies: optical motion capture, pressure distribution measurement (pressure platform, foot insoles, etc.), combined technologies, etc.

In recent years, inertial measurement units (IMUs, inertial sensors) containing accelerometers, gyroscopes, and often magnetometers started to be applied for the gait analysis [5–10]. Their advantages in comparison with other technologies are the following: miniature size, lightweight, and portability; can be easily used indoors and outdoors; ability

to record the motion of any part of the body (arm, spine, head, etc.); and low-cost IMU chips that make this kind of systems affordable.

The development of a new clinical gait analysis system using inertial sensors was the ultimate goal of the authors of this article. For this purpose, we have created new Neurosens inertial sensors containing a 3D (three dimensional) accelerometer и 3D gyroscope. We also developed a new algorithm for heel-strike (HS) and toe-off (TO) event detection based on deep neural networks.

The following requirements were imposed on the developed system. First of all, the correct sensor placement should be possible in the patient with various anatomical disorders.

The sensor positioning should maximally prevent sensor movements caused by the displacement of soft tissues. Secondly, the algorithm for HS and TO event detection should easily be modified for new groups of patients with various types of movement pathologies.

Different ways of sensor positioning are known in the literature, and sensors can be placed from the head [11] to the back of the foot [12]. In most cases, a two-sensor configuration is used when both sensors are positioned on the feet [5, 8, 13–15]. Another way is to place sensors on the shanks. Shank-worn sensors show less signal variability between subjects if compared to the foot-worn sensors [16]. Special footwear or footwear accessories are not required in this case that reduces ready-for-operation time. Additionally, shank placement provides accurate measurements and event detection in healthy subjects [17, 18] and in case of gait impairments during level ground walking [18–20].

In the case of our system, we decided to place sensors on lateral malleoli because of the anatomical narrowing in this area; the sensors can be securely fixed. Stability is achieved by placing the sensors directly on the wide bone surface of the lateral malleolus; such positioning also helps to minimize sensor displacement caused by soft tissue movements. At the same time, nothing interferes with the movements in the ankle joint. Another reason for this sensor placement is that a lot of patients (for example, with cerebral palsy) suffer from foot deformity, which prevents stable fixation of sensors on the feet. Besides, foot movements in such patients can be chaotic and uncontrolled that can reduce the performance of the HS and TO detection algorithm.

To detect HS and TO events, we decided to use the algorithm based on deep neural networks. There are many algorithms for HS and TO detection based on local extrema determination of gyroscope and/or accelerometer data using thresholding techniques [5, 19, 21–23]. However, in case of poor performance of a deterministic algorithm in a new patient group, it may be necessary to modify the entire algorithm up to its complete replacement. Whereas for an algorithm based on neural networks, a simple retraining of the neural network can be sufficient, and in our opinion, this is much less expensive than modifying a deterministic algorithm.

We could not find the HS and TO event detection algorithm based on neural networks in the case of shank-worn sensor placement in the literature. In [9, 23], sensors are placed on lateral malleolus, but thresholding algorithms are used. In [24, 25], neural networks are used for HS and TO event detection but sensor placement differs from lateral malleolus. Wang et al. [26] proposed a recurrent neural network (RNN) that detects HS event (but not TO event) with sensors placed on the lateral malleolus. Using the same sensor placement, Sarshar et al. [10] designed RNN for the detection of foot-contact (the time point when the foot contacts the floor) and foot-off (the time point when the foot leaves the ground) events. These events are time-shifted compared to HS and TO events. Therefore, we decided to develop a new algorithm for HS and TO event detection.

In summary, our main contributions are the development of a new neural network algorithm for HS and TO

detection in the case when IMU sensor placement is on lateral malleolus, as well as the validation of a gait analysis system that uses the proposed HS and TO event detection algorithm and new Neurosens IMU sensors.

2. Materials and Methods

2.1. Sensor Data. The Neurosens wireless sensors (produced by Neurosoft, Russia) were used to obtain the gait data. Technical specifications of the sensors are the following: 3D accelerometer (output frequency: 200 Hz, range: ± 16 g), 3D gyroscope (output frequency: 200 Hz, range: $\pm 2000^\circ/\text{s}$), and wireless interface: Wi-Fi.

The process of data transmission from the sensor to the personal computer is carried out using a Wi-Fi router. Steadys software (Neurosoft, Russia) was used for data collecting, processing, and sensor synchronization (synchronization error < 5 ms).

2.2. Sensor Placement. In this paper, we considered the case of sensors placed on shanks. Namely, the sensors were placed on the lateral malleolus and fixed using elastic straps as shown in Figure 1.

2.3. Reference System. The RehaWalk® (Zebris Medical GmbH, Germany) for stance and gait analysis was used as a reference system. The system consisted of a capacitance-based pressure platform housed within a treadmill. The pressure platform had a sensing area of 108.4×47.4 cm and incorporated 7,168 sensors, each approximately 0.85×0.85 cm. The treadmill had a contact surface of 150×50 cm, and its belt speed could be adjusted between 0.2 and 22 km/h, at intervals of 0.1 km/h. Data from the pressure platform was transmitted to the computer via a USB interface. The sampling rate was 200 Hz. Zebris Measurement Suit software version 1.18 was used to obtain measurements of gait parameters.

2.4. Subjects and Trials. Healthy volunteers who took part in the study were divided into two groups. The first group was used to get the main dataset for deep neural network training. This group included 60 subjects (30 males and 30 females, aged 18–62 years, height 153–192 cm). The second group was used to get the holdout dataset (this dataset was not used for network training in any way), which was used to compare the accuracy and precision of the proposed and reference systems. This group included 117 subjects (62 males and 55 females, aged 24–55 years, height 162–183 cm). All participants gave written informed consent prior to participation in the study. The study received approval from the local research ethics committee.

The participants walked with different speeds listed in Table 1.

Every subject was walking on the RehaWalk® treadmill system (reference system). Two to five minutes were given for every subject to adapt to the treadmill speed. After that, gait parameters were recorded using the reference and proposed systems at the same time. Each trial continued for two minutes.



FIGURE 1: Sensor placement on the shanks using elastic straps.

TABLE 1: Walking speeds.

Walking speed, km/h	First group (main dataset)	Second group (holdout dataset)
0.5	8 men and 7 women	15 men and 14 women
2	7 men and 8 women	16 men and 14 women
3	8 men and 7 women	16 men and 13 women
5	7 men and 8 women	15 men and 14 women

2.5. Main Dataset Collection. The following steps were performed at every trial in the first group to get the main dataset: (1) raw data (accelerometers and gyroscopes data) of each sensor were saved; (2) information about HS and TO events was extracted from the RehaWalk® treadmill system using “Export to XML” function; (3) the sensor data and reference data were synchronized via the synchronization input/output interface of the RehaWalk® treadmill system.

3. Methods

3.1. HS and TO Event Detection Algorithm. We developed three models based on deep neural networks for HS and TO detection: HS and TO initial detection model, HS refinement model, and TO refinement model. Models were applied for each leg separately. Neural networks were written in Python 3.6.5 with TensorFlow 1.9.0 [27] and Keras 2.2.0 [28] libraries.

3.1.1. HS and TO Initial Detection Model. This model is a convolutional neural network which consists of 8 convolutional layers, 2 input layers (1 to 6 convolutional layers: first input branch; 7 to 8 convolutional layers: second input branch), and 1 output layer (Figure 2(a)). The next statements are applied for each convolutional layer of a proposed network: activation function is ReLU [29] which is defined as $f(x) = \max(0, x)$; number of filters is equal to 6; kernel size is equal to 2; stride is equal to 1.

Dilated convolution [30] is used from 2 to 5 layers (first input branch) with dilation rate of 2^{n-1} , where n is the layer number. Dilated convolution is defined as the following:

Let $F : \mathbb{Z}^2 \rightarrow \mathbb{R}$ be a discrete function. Let $\Omega_r = [-r, r]^2 \cap \mathbb{Z}^2$ and let $k : \Omega_r \rightarrow \mathbb{R}$ be a discrete filter of size $(2r + 1)^2$. The discrete convolution operator $*$ can be defined as

$$(F * k)(p) = \sum_{s+t=p} F(s)k(t), \quad (1)$$

$$(F *_l k)(p) = \sum_{s+lt=p} F(s)k(t).$$

Here, $*_l$ is an l -dilated convolution. Common discrete convolution $*$ is simply the 1 dilated convolution.

For kernels, initialization Glorot uniform initializer [31] is used. It is defined as

$$\mathcal{U}\left(-\frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}, \frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}\right), \quad (2)$$

where \mathcal{U} is a random uniform distribution, n_i is the number of incoming connections or “fan-in” to the layer, and n_{i+1} is the number of outgoing connections from that layer, also known as the “fan-out.”

The network analyzes data segments of 60 timestep (points in time) duration which overlap by 20 timesteps. This duration was chosen because we wanted to give the network as much temporal information as possible, but at the same time, we did not want to confuse the network during the training by feeding the segments where both HS and TO events occurred. In fact, the network makes a decision only for 20 timesteps in the middle of this segment, and the rest of the timesteps are used as additional information. 20 timesteps were chosen because they represent the

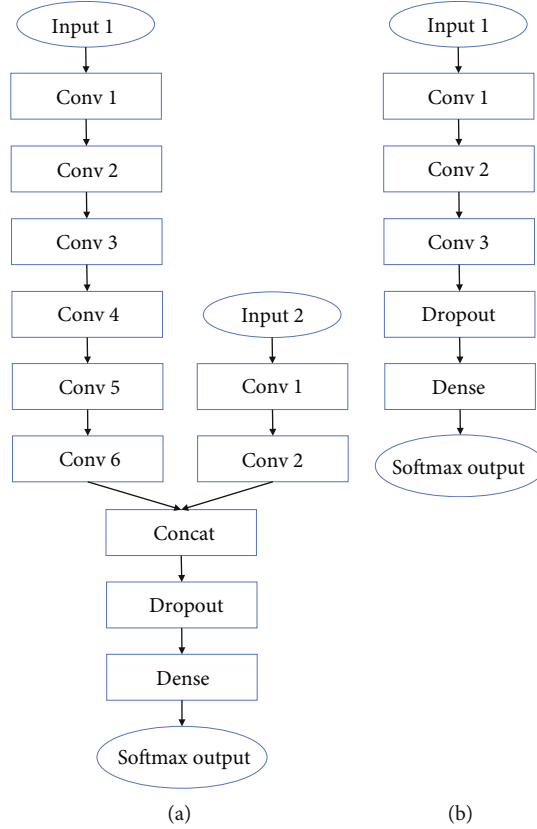


FIGURE 2: (a) HS and TO initial detection model architecture. (b) HS and TO refinement model architecture.

duration that is a little bit less than the minimal time between HS and TO in our dataset.

The output of the last convolutional layer is transferred to a dense layer with an output size equal to 3. For weight initialization of this layer, a uniform initializer is used which is defined as $\mathcal{U}(-0.05, 0.05)$, where \mathcal{U} is a random uniform distribution.

Then, the output of this layer is transferred to the last layer of the network.

The last layer of the network is a softmax [32] function which is defined as follows:

Let $\sigma : \mathbb{R}^K \rightarrow (0, 1)^K$ is defined when $K \geq 1$; then

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_j^K e^{z_j}}, \quad (3)$$

where $z = (z_1, \dots, z_K) \in \mathbb{R}^K$ and K is the number of classes (equals to 3).

This layer produces a three-dimensional output vector for each analyzed data segment which represents probability distribution over three possible classes, such as (1) the data segment contains HS, (2) the data segment contains TO, and (3) the data segment does not contain HS and TO.

Dropout [33] (with dropout rate equal to 0.1) and L2-regularization [34] (with regularization factor λ equal to 10^{-4}) are used to prevent overfitting. L2-regularization is defined as

$$E_{L2}(w_1, \dots, w_l) = E(w_1, \dots, w_l) + \lambda \sum_{i=1}^l \|w_i\|^2. \quad (4)$$

Cross entropy is used as a loss function and defined as

$$CE = - \sum_{i=1}^C t_i \log(f(s)_i), \quad (5)$$

where C is the number of classes (equals to 3), t is the target vector, and $f(s)$ is an output vector of the network.

Parameters of the HS and TO initial detection model are given in Table 2.

Input and output dimensions of each respective layer of the network are fully defined by layer parameters described above and dimensions of input data described below.

Data preprocessing for the first input layer: The data segment is fed to the first input layer as a matrix 6×60 which consists of three axis accelerometers and three axis gyroscope time sequences (sequences of values over time), where the rows correspond to the time sequences and the columns to the timesteps (Figure 3). For each row x min-max normalization from zero to one is applied, which is defined as $x_{\text{normalized}} = (x - x_{\min}) / (x_{\max} - x_{\min})$.

Data preprocessing for the second input layer: The data fed to the second input layer is a matrix 6×9 which consists of Pearson's correlation coefficients, which are defined as follows.

TABLE 2: Parameters of the detection model layers.

Layer	Type	Dimensions	Dilation rate	Filters	Kernel size	Stride
Input 1	Input	$N \times 60 \times 6$	—	—	—	—
Conv 1 (input 1)	Convolutional	$2 \times 6 \times 6$	1	6	2	1
Conv 2 (input 1)	Convolutional	$2 \times 6 \times 6$	2	6	2	1
Conv 3 (input 1)	Convolutional	$2 \times 6 \times 6$	4	6	2	1
Conv 4 (input 1)	Convolutional	$2 \times 6 \times 6$	8	6	2	1
Conv 5 (input 1)	Convolutional	$2 \times 6 \times 6$	16	6	2	1
Conv 6 (input 1)	Convolutional	$2 \times 6 \times 6$	1	6	2	1
Input 2	Input	$N \times 9 \times 6$	—	—	—	—
Conv 1 (input 2)	Convolutional	$2 \times 6 \times 6$	1	6	2	1
Conv 2 (input 2)	Convolutional	$2 \times 6 \times 6$	1	6	2	1
Dense (concatenate)	Dense	414×3	—	—	—	—
Output	Softmax	3	—	—	—	—

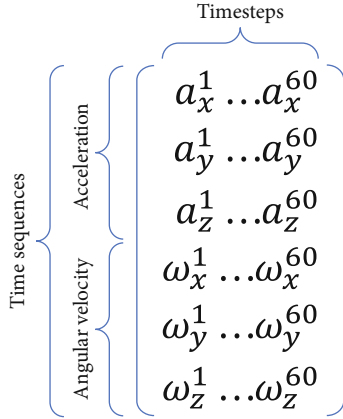


FIGURE 3: Matrix fed to the first input layer.

Given paired data $\{(x_1, y_1), \dots, (x_n, y_n)\}$ consisting of n pairs (n is equal to 60), r_{xy} is defined as

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}, \quad (6)$$

where \bar{x}, \bar{y} are average values of corresponding vectors.

These coefficients are calculated as follows: the current matrix for the first input layer and the last three matrices of each class (total of nine matrices) from the matrices previously fed to the first input layer and already classified by the model are used. Pearson correlation coefficients are calculated between the rows of current matrix and corresponding rows of nine matrices (Figure 4). This information is used as a network memory and our experiments showed that it helped to increase HS and TO detection robustness.

3.1.2. HS Refinement Model. This model is a convolutional neural network which consists of 3 convolutional layers, 1 input layer, and 1 output layer (Figure 2(b)). The next statements are applied for each convolutional layer of a proposed

network. The activation function is PReLU [35] which is defined as follows:

$$f(x) = \begin{cases} x, & \text{if } x > 0, \\ ax, & \text{otherwise} \end{cases}, \quad (7)$$

where parameter a is learned along with the other neural network parameters. The number of filters is equal to 12, kernel size is equal to 6, and stride is equal to 1.

Dilated convolution is used from second to third layers with a dilation rate of 2^{n-1} , where n is the layer number. Glorot uniform initializer is used for kernel initialization. The data fed to the first input layer of the initial detection model and classified by it as the first class is used as an input data for this refinement model. The output of the last convolutional layer is transferred to a dense layer with an output size equal to 20. For weight initialization of this layer, a uniform initializer is used which is defined as $\mathcal{U}(-0.05, 0.05)$, where \mathcal{U} is a random uniform distribution. Then, the output of this layer is transferred to the last layer of the network.

Softmax function is used as the last network layer, and it yields a vector which represents probability distribution over 20 classes, and each class corresponds to the timestep inside the middle 20 timesteps of an input data segment when the HS event occurred.

Dropout (with dropout rate equal to 0.2) and L2-regularization (with regularization factor λ equal to 10^{-4}) are used to prevent overfitting. Cross entropy is used as a loss function.

Parameters of the HS refinement model are given in Table 3.

Input and output dimensions of each respective layer of the network are fully defined by layer parameters and dimensions of input data described above.

3.1.3. TO Refinement Model. The architecture of this model is the same as the architecture of the model described in Section 3.1.2. This model is applied only to those data segments

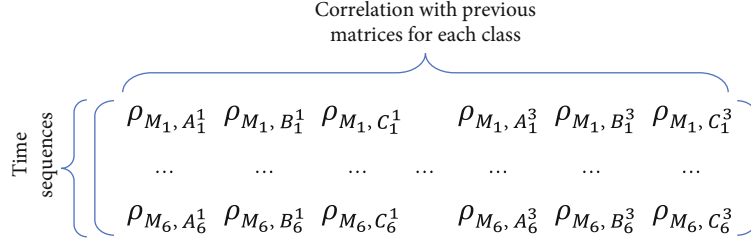


FIGURE 4: Matrix fed to the second input layer. M : current matrix for the first input layer (lower index is a row index); $A^1 - A^3$: last three matrices of the first class; $B^1 - B^3$: last three matrices of the second class; $C^1 - C^3$: last three matrices of the third class.

TABLE 3: Parameters of the HS refinement model layers.

Layer	Type	Dimensions	Dilation rate	Filters	Kernel size	Stride
Input 1	Input	$N \times 60 \times 6$	—	—	—	—
Conv 1	Convolutional	$6 \times 6 \times 12$	1	12	6	1
Conv 2	Convolutional	$6 \times 12 \times 12$	2	12	6	1
Conv 3	Convolutional	$6 \times 12 \times 12$	4	12	6	1
Dense	Dense	720×20	—	—	—	—
Output	Softmax	20	—	—	—	—

that were classified by the initial detection model as second class.

3.1.4. Model Training. For network training, our main dataset divided patient-wise into two datasets: validation set (20 patients) and remaining set (40 patients). We used a five-fold cross-validation scheme. The remaining set was divided into five folds in a semirandom way so that each fold contained the data acquired from four men and four women for each walking speed. At each iteration, one fold was selected as a test set, and the remaining four folds were used as a training set. This process was repeated five times. At each iteration, the detection model was trained over 300 epochs, and each refinement model was trained over 2500 epochs. The best model was chosen at each iteration, and then, these models were averaged over all iterations to assess the final model performance. The validation set was used for tuning hyperparameters. The networks were trained using Nesterov Adam optimizer [36] with batch size of 256 fragments, learning rate equal to 0.002, β_1 equal to 0.9, and β_2 equal to 0.999.

Figure 5 shows the entire workflow of the proposed algorithm in case of heel-strike detection (toe-off detection is performed in the same way).

4. Results

4.1. HS and TO Detection. To assess the final detection model performance, sensitivity and specificity were calculated. Each record from holdout dataset (this dataset was not used for networks training in any way) was divided into series of 0.1 second (20 timesteps) segments. Then, for each segment, it was determined how model performed true-positive (TP, where model output was HS or TO, and annotation was HS or TO, respectively), true-negative (TN, where

model output was neither HS nor TO, and annotation was neither HS nor TO, respectively), false-positive (FP, where model output was HS or TO while annotation was neither HS nor TO, respectively), and false-negative (FN, where model output was neither HS nor TO while annotation was HS or TO, respectively). Then, the total count of each TP, TN, FP, FN was calculated over the entire dataset. Then, sensitivity (SE) and specificity (SP) were calculated as follows:

$$\begin{aligned}
 SE &= \frac{TP}{TP + FN} * 100\%, \\
 SP &= \frac{TN}{TN + FP} * 100\%.
 \end{aligned} \tag{8}$$

Results for HS and TO can be seen in Table 4.

To assess final refinement model performance, threshold accuracy was calculated. Each record from the holdout dataset (this dataset was not used for network training in any way) was divided into a series of 0.1 second (20 timesteps) segments. Then, only segments where the annotation of HS or TO was present were selected. Then, threshold accuracy (TA) with window of 25 milliseconds for HS and TO, respectively, was calculated as follows:

$$TA = \frac{T_{25\text{ms}}}{N} * 100\%, \tag{9}$$

where N is a total number of segments where the annotation of HS or TO was present over entire dataset, and $T_{25\text{ms}}$ is the number of segments from N segments for which $|\text{argmax}(t) - \text{argmax}(r)| < 5$ is true over the entire dataset (5 time-steps equal to 25 milliseconds), where t is the output vector of the refinement model and r is the reference vector.

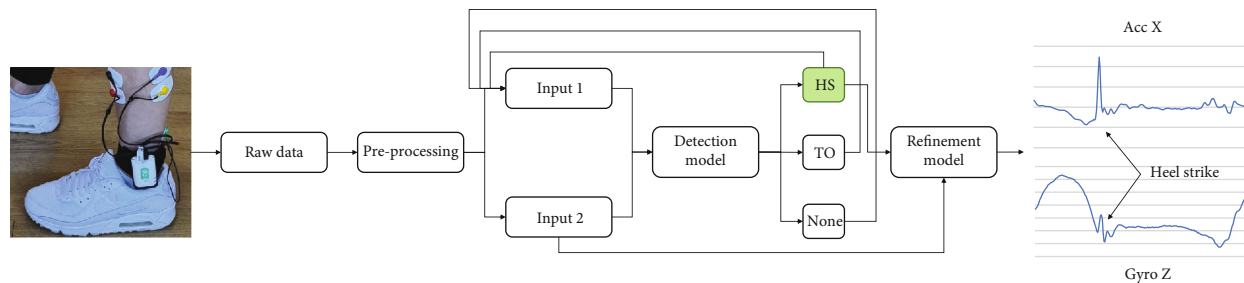


FIGURE 5: Algorithm workflow in case of determining the moment of heel strike.

TABLE 4: Sensitivity, specificity, and threshold accuracy of models.

Parameter	HS	TO
SE, %	98, 74	97, 68
SP, %	98, 90	99, 17
TA, %	97, 77	95, 61

Resulting values of TA for HS refinement model and TO refinement model are listed in Table 4.

4.2. Validation of the Gait Analysis System. To validate the proposed Steadys gait analysis system, the following mean values of gait parameters were calculated for each record from the holdout dataset: gait cycle, step, cadence, gait phases (stance phase, swing phase, single support, double support, load response, and preswing). The gait parameters were calculated according to Winter [2]. Average values were calculated over the whole trial duration.

Then, we calculated the absolute difference (ϵ) between the reference and proposed systems for each gait parameter of every subject. Accuracy (mean of ϵ) and precision (standard deviation, STD of ϵ) for each estimated gait parameter are given in Table 5. Along with absolute difference, we calculated relative difference (rel. ϵ) where each difference value ϵ was divided by correspondent reference value and expressed as a percentage.

5. Discussion

As it was mentioned in the introduction, we could not find the existing HS and TO event detection algorithms based on neural networks in the case of shank-worn sensor placement in the literature. Therefore, a direct comparison of the accuracy of the proposed algorithm and the gait analysis system based on this algorithm is difficult to conduct. In addition, all these studies differ in sampling of study subjects. Thus, [9] tested healthy subjects, [19] included the study dataset coxarthrosis subjects and hip-arthroplasty patients, and [26] focused on healthy elderly subjects, stroke subjects, and subjects with Parkinson's disease. Also, different researches used different HS and TO event detection criteria. For instance, [9] applied the optoelectronic motion capture system to determine reference events, [19] used the force plate, and [26] found reference events by the gyroscope signal from the sensors positioned on the shanks.

TABLE 5: Differences and relative differences between the reference and proposed systems.

Parameter	Mean \pm std of ϵ	Mean of rel. ϵ , %
Gait cycle, s	-0.021 ± 0.091	-0.891 ± 3.843
Step, s	-0.008 ± 0.041	-0.563 ± 3.454
Cadence, steps/min.	0.589 ± 1.144	1.127 ± 2.748
Stance phase, % of gait cycle	-0.051 ± 0.544	-0.070 ± 0.734
Single support, % of gait cycle	-0.370 ± 0.649	-1.518 ± 2.851
Load response, % of gait cycle	0.132 ± 0.561	0.569 ± 2.541
Preswing, % of gait cycle	0.106 ± 0.661	0.459 ± 2.945
Double support, % of gait cycle	0.296 ± 0.711	0.660 ± 1.622
Swing phase, % of gait cycle	0.056 ± 0.578	0.207 ± 2.487

However, we compared our algorithm with other algorithms, in which sensors are placed on shanks. Accuracy for gait cycle and cadence is similar to [9]; Renggli et al. reported absolute errors 0.0 ± 0.03 s for gait cycle and -0.9 ± 4.5 steps/min. for cadence. The results of the proposed system are slightly worse compared to Salarian et al. [19], e.g., Salarian et al. obtained error 0.0022 ± 0.0232 s for gait cycle.

Mariani et al. [8] used sensors on the feet and reported errors for load response, single support, and preswing expressed as percentage of stance phase. Correspondent errors for our system expressed as percentage of stance phase are slightly lower.

Among the alternative algorithms for HS and TO event detection based on neural networks, Wang et al. [26] got slightly better sensitivity for detection of HS events with more than 99.65% value. Such high value can be partly explained by the fact that Wang et al. defined reference HS events using gyroscope signals that were also used during neural network training.

The study design, as well as the parameters analyzed by the authors, differed in each of the above mentioned studies, which makes direct comparison not always possible. In general, we can conclude that the performance of the proposed algorithm is similar to other state-of-the-art algorithms. At the same time, the obtained differences in accuracy and precision in other studies compared to the proposed algorithm are not significant from a clinical

point of view due to the insignificance of the absolute values of obtained errors.

It should be mentioned that the proposed algorithm has some limitations, such as simultaneous use of 3D accelerometers and 3D gyroscopes is mandatory. Modifying models to use data from only a single sensor (e.g., accelerometer) or with fewer degrees of freedom (e.g., 1D or 2D) is possible but could lead to much worse performance since full reconstruction of the motion is not possible without 3D data. Another limitation is that the algorithm supports only the 200 Hz frequency. Modifying the models to support other frequencies is possible, although lower frequency will directly decrease algorithm accuracy. And the last one relates to search limitations of the algorithm. The algorithm performs search only for heel-strikes and toe-offs of respective leg where the sensor is attached (e.g., left or right, respectively). Thus, to calculate some of the resulting gait parameters (e.g., double support), usage of 2 sensors (on the left and right leg) is mandatory.

At the same time, the described algorithm has no significant limitations related to the place of sensor fixation. For example, the main dataset can be used, in which the sensors were attached to the feet. There are no limitations regarding various pathological gait patterns in the subject of the study or leg amputations above the ankle joint (if there is prosthesis). However, the algorithm performance in all these cases requires additional study.

6. Conclusion

In this paper, we introduced the new neural network algorithm for HS and TO event detection using wearable IMU sensors placed on lateral malleolus.

We used the proposed HS and TO event detection algorithm and new Neurosens IMU sensors to build the new gait analysis system.

In the present paper, we evaluated the performance of both the new algorithm and the new gait analysis system, and it turned out to be comparable with other state-of-the-art algorithms.

We hope that this article will be helpful for all specialists who are involved in the development of clinical gait analysis systems.

Data Availability

No primary data is attached to the article. Primary data and other materials can be requested from the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] V. T. Inman, H. J. Ralston, and F. Todd, *Human Walking*, Williams & Wilkins, Baltimore, London, UK, 1st ed. edition, 1981.
- [2] D. A. Winter, *The Biomechanics and Motor Control of Human Gait*, University of Waterloo Press, Waterloo, Ontario, Canada, 2nd ed. edition, 1991.
- [3] J. Perry, *Gait Analysis: Normal and Pathological Function*, Thorofare, NJ, USA, SLACK Incorporated, 1st ed. edition, 1992.
- [4] C. Kirtley, *Clinical Gait Analysis: Theory and Practice*, Churchill Livingstone, London, UK, 1st ed. edition, 2006.
- [5] A. M. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, "Assessment of walking features from foot inertial sensing," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 3, pp. 486–494, 2005.
- [6] B. Mariani, C. Hoskovec, S. Rochat, C. Bula, J. Penders, and K. Aminian, "3D gait assessment in young and elderly subjects using foot-worn inertial sensors," *Journal of Biomechanics*, vol. 43, no. 15, pp. 2999–3006, 2010.
- [7] S. Chen, J. Lach, B. Lo, and G. Z. Yang, "Toward pervasive gait analysis with wearable sensors: a systematic review," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 6, pp. 1521–1537, 2016.
- [8] B. Mariani, H. Rouhani, X. Crevoisier, and K. Aminian, "Quantitative estimation of foot-flat and stance phase of gait using foot-worn inertial sensors," *Gait & Posture*, vol. 37, no. 2, pp. 229–234, 2013.
- [9] D. Renggli, C. Graf, N. Tachatos et al., "Wearable inertial measurement units for assessing gait in real-world environments," *Frontiers in Physiology*, vol. 11, p. 90, 2020.
- [10] M. Sarshar, S. Polturi, and L. Schega, "Gait phase estimation by using LSTM in IMU-based gait analysis—proof of concept," *Sensors*, vol. 21, no. 17, p. 5749, 2021.
- [11] T. H. Hwang, J. Reh, A. O. Effenberg, and H. Blume, "Real-time gait analysis using a single head-worn inertial measurement unit," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 2, pp. 240–248, 2018.
- [12] A. Küderle, N. Roth, J. Zlatanovic, M. Zrenner, B. Eskofier, and F. Kluge, "The placement of foot-mounted IMU sensors does affect the accuracy of spatial parameters during regular walking," *PLoS One*, vol. 17, no. 6, article e0269567, 2022.
- [13] A. R. Anwary, H. Yu, and M. Vassallo, "Optimal foot location for placing wearable IMU sensors and automatic feature extraction for gait analysis," *IEEE Sensors Journal*, vol. 18, no. 6, pp. 2555–2567, 2018.
- [14] S. N. Ghousayni, J. R. Henty, D. E. Wood, and D. J. Ewins, "An experimental evaluation of the gyroscope as a sensor in FES foot-drop correction systems," in *Proceedings of the 7th Vienna International Workshop on Functional Electrical Stimulation*, pp. 200–203, Vienna, Austria, 2001.
- [15] M. Zago, M. Tarabini, M. Delfino Spiga et al., "Machine-learning based determination of gait events from foot-mounted inertial units," *Sensors*, vol. 21, no. 3, p. 839, 2021.
- [16] G. Wu, "A review of body segmental displacement, velocity and acceleration in human gait," in *Gait Analysis Theory and Application*, R. L. Craik and C. A. Oatis, Eds., pp. 205–222, Mosby, St Louis, MO, USA, 1995.
- [17] K. Aminian, B. Najafi, C. Bula, P. F. Leyvraz, and P. Roberts, "Spatio-temporal parameters of gait measured by an ambulatory system using miniature gyroscopes," *Journal of Biomechanics*, vol. 35, no. 5, pp. 689–699, 2002.
- [18] S. N. Ghousayni, *Application of Angular Rate Gyroscopes as Sensors In Electrical Orthoses for Foot Drop Correction*, [Ph.D. thesis], University of Surrey, Guildford, UK, 2004.

- [19] A. Salarian, H. Russmann, F. J. G. Vingerhoets et al., "Gait assessment in Parkinson's disease: toward an ambulatory system for long-term monitoring," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 8, pp. 1434–1443, 2004.
- [20] K. Aminian, C. Trevisan, B. Najafi et al., "Evaluation of an ambulatory system for gait analysis in hip osteoarthritis and after total hip replacement," *Gait & Posture*, vol. 20, no. 1, pp. 102–107, 2004.
- [21] P. Catalfamo, S. Ghousayni, and D. Ewins, "Gait event detection on level ground and incline walking using a rate gyroscope," *Sensors*, vol. 10, no. 6, pp. 5683–5702, 2010.
- [22] J.-K. Lee and E. J. Park, "Quasi real-time gait event detection using shank-attached gyroscopes," *Medical & Biological Engineering & Computing*, vol. 49, no. 6, pp. 707–712, 2011.
- [23] P. Fraccaro, L. Walsh, J. Doyle, and D. O'Sullivan, "Real-world gyroscope-based gait event detection and gait feature extraction," in *Proceedings of The 6th International Conference on eHealth, Telemedicine, and Social Medicine*, pp. 247–252, Barcelona, Spain, 2014.
- [24] J. Hannink, T. Kautz, C. F. Pasluosta, K.-G. Gaßmann, J. Klucken, and B. M. Eskofier, "Sensor-based gait parameter extraction with deep convolutional neural networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 1, pp. 85–93, 2017.
- [25] M. Z. Arshad, A. Jamsrandorj, J. Kim, and K.-R. Mun, "Gait events prediction using hybrid CNN-RNN-based deep learning models through a single waist-worn wearable sensor," *Sensors*, vol. 22, no. 21, p. 8226, 2022.
- [26] F.-C. Wang, Y.-C. Li, T.-Y. Kuo, S.-F. Chen, and C.-H. Lin, "Real-time detection of gait events by recurrent neural networks," *IEEE Access*, vol. 9, pp. 134849–134857, 2021.
- [27] M. Abadi, P. Barham, J. Chen et al., "TensorFlow: a system for large-scale machine learning," 2016, <https://arxiv.org/abs/1605.08695>.
- [28] "Keras: The Python Deep Learning library," 2019, <https://keras.io>.
- [29] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 315–323, Fort Lauderdale, FL, USA, 2011.
- [30] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2016, <https://arxiv.org/abs/1511.07122v3>.
- [31] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, Chia Laguna Resort, Sardinia, Italy, 2010.
- [32] J. S. Bridle, "Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition," in *Neurocomputing: Algorithms, Architectures and Applications*, F. Fogelman Soulie and J. Hertz, Eds., pp. 227–236, Springer-Verlag, Berlin, Heidelberg, Germany, 1990.
- [33] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, <https://arxiv.org/abs/1207.0580v1>.
- [34] A. Y. Ng, "Feature selection, L_1 vs. L_2 regularization, and rotational invariance," in *ICML '04 Proceedings of the twenty-first international conference on Machine learning*, p. 78, Banff, Alberta, Canada, 2004.
- [35] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-Level performance on ImageNet classification," *IEEE International Conference on Computer Vision (ICCV 2015)*, vol. 1502, pp. 1026–1034, 2015.
- [36] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on Machine Learning*, pp. 1139–1147, Atlanta, GA, USA, 2013.