sensors

Special Issue Reprint

Robust Motion Recognition Based on Sensor Technology

Edited by
Manuel Gil-Martín, Rubén San-Segundo and Fernando Fernández-Martínez

mdpi.com/journal/sensors

MDPI

Robust Motion Recognition Based on Sensor Technology

Robust Motion Recognition Based on Sensor Technology

Guest Editors

Manuel Gil-Martín
Rubén San-Segundo
Fernando Fernández-Martínez

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editors
Manuel Gil-Martín
Department of
Electronics Engineering
Universidad Politécnica
de Madrid
Madrid
Spain

Rubén San-Segundo
Department of
Electronics Engineering
Universidad Politécnica
de Madrid
Madrid
Spain

Fernando
Fernández-Martínez
Department of
Electronics Engineering
Universidad Politécnica
de Madrid
Madrid
Spain

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Sensors* (ISSN 1424-8220), freely accessible at: https://www.mdpi.com/journal/sensors/special_issues/3U25798XE3.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-3675-8 (Hbk)
ISBN 978-3-7258-3676-5 (PDF)
https://doi.org/10.3390/books978-3-7258-3676-5

Contents

sensors

Article

Designing a Multivariate Belt Conveyor Idler Stall Detection and Identification System with Scalability Analysis

Kyeong Su Shin [1], Younho Nam [2] and Young-Joo Suh [1,*]

[1] Graduate School of Artificial Intelligence, Pohang University of Science and Technology, Pohang 37673, Republic of Korea; ksshin@postech.ac.kr
[2] Department of Computer Science and Engineering, Pohang University of Science and Technology, Pohang 37673, Republic of Korea; younho@postech.ac.kr
* Correspondence: yjsuh@postech.ac.kr; Tel.: +82-54-279-2243

Abstract: Belt conveyor idlers are freely rotating idlers supporting the belt of a conveyor, and can induce severe frictional damage to the belt as they fail. Therefore, fast and accurate detection of idler faults is crucial for the effective maintenance of belt conveyor systems. In this article, we implement and evaluate the performance of an idler stall detection system based on a multivariate deep learning model using accelerometers and microphone sensor data. Emphasis is place on the scalability of the system, as large belt conveyor installations can span multiple kilometers, potentially requiring hundreds or even thousands of sensor units to monitor. The accuracy of the proposed system are analyzed and reported, along with its network bandwidth and energy requirements. The results suggest that while implementing accurate large-scale idler stall detection is feasible, careful consideration must be paid to observing the available network bandwidth and energy budget in order to avoid prolonged downtimes.

Keywords: fault diagnosis; anomaly detection; belt conveyor idler; machine learning; wireless sensor networks; scalability

Citation: Shin, K.S.; Nam, Y.; Suh, Y.-J. Designing a Multivariate Belt Conveyor Idler Stall Detection and Identification System with Scalability Analysis. *Sensors* **2024**, *24*, 7989. https://doi.org/10.3390/s24247989

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 13 November 2024
Revised: 11 December 2024
Accepted: 12 December 2024
Published: 14 December 2024

1. Introduction

Conveyor belt idlers are cylindrical freely-rotating supports installed along the path of a belt conveyor to support the belt and prevent it from sagging. Idler function is critical to the lifespan of the belt, as the idlers reduce the wear and tear experienced by the belt by supporting it and preventing any drag that could be induced by other structures. Idler failures caused by fault conditions such as failed bearings can cause the idler to stall and drag against the belt. Drag on the belt caused by stalled idlers can induce frictional damage, to the point where the belt or even the whole conveyor system fails (Figure 1).

There have been multiple attempts to implement a system that can automatically detect or predict the idler failures using various sensing technologies. Review articles from Liu and Alharbi [1,2] summarize the current progress of such systems. Earlier designs were often based on the use of thermometers or thermal cameras to identify faults by detecting temperature increases caused by increased friction between the belt and failed idler [3,4]. However, such designs have relatively low sensitivity, and can only detect failure conditions at relatively late stages. More advanced designs often employ vibration sensors, microphones, or acoustic emission sensors along with machine learning (ML) classifiers to detect the vibrational patterns of failed idlers [1,2] (example: [5,6]). Such systems operate by collecting the raw vibration/acoustic sample data from single or multiple sensor units, preprocessing them (usually into spectrograms, wavelet plots, or handcrafted features), then feeding the preprocessed data into ML classifiers. Most works have concentrated on improving the feature extractor and/or model design; for example, a recent manuscript from Alharbi [5] achieved improved detection accuracy by utilizing a pretrained model and an Long Short-Term Memory (LSTM)-based feature extractor.

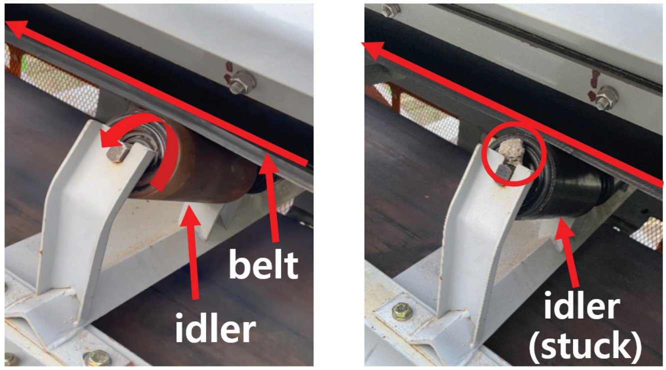


Figure 1. A belt idler moving freely (**left**) and intentionally stalled for depiction and data collection purposes (**right**). The red arrows on the belt and the idler indicate the direction of the movements. Other arrows, circles, and texts are simply used to emphasize important components of the system.

Unfortunately, due to the challenges and costs associated with installing and maintaining such systems, most real-world belt conveyor operators do not implement such systems, instead simply relying on manual inspections. This is largely due to the scale of belt conveyor installations, with manual inspection usually sufficient for small belt conveyors. For larger installations, such as those seen in mines, steelworks, and power stations, the length of the belt conveyor can span multiple kilometers [7], requiring hundreds or even several thousand sensor units to be installed and maintained. Moreover, these sensor units are often exposed to harsh conditions due to the nature of the tasks involved, and may not have access to high-bandwidth networks or mainline electricity. This makes sensor installation and maintenance challenging tasks, to the point where operators may simply choose not to implement such a system and resort to traditional manual inspection techniques.

This suggests to us that the scalability of these systems needs to be studied and improved in order to allow for wider adaptation of new proposed technologies. Until now, most works have concentrated on improving the accuracy of detection systems, leaving the scalability of these systems somewhat underexplored. The following problems must be solved in order to enable wider adaptation of such technologies: resiliency of the system against sensor failures, network bandwidth requirements, and energy requirements of the system. Because hundreds or thousands of sensor units are exposed to harsh conditions, some sensor failures are expected; thus, the detection system must be able to cope with sensor losses. In addition, if the detection system is based on multivariate sensors, such as accelerometers or acoustic sensors with relatively high sampling rates, then the network bandwidth and energy requirements of the sensor units can grow prohibitively high as the system scales. Thus, the system must be designed and analyzed with the network and energy budgets in mind.

As the size of the system grows, the chance of sensor failure increases. Intuitively, the chance of not experiencing a sensor failure at all at a given moment is x^n, where n is the number of the sensor units and $x < 1$ is the availability of the sensor at that moment (assuming independent and identically distributed probabilities). As the number of sensors grows, this probability decreases rapidly. There have been several studies on diagnosing sensor faults and mitigating the effects of failures [8–10]. These articles mainly concentrate on topics such as detecting sensor failures themselves [8], identifying typical failure modes, and implementing rule-based or machine learning-based sensor failure detection models, while some other articles concentrate on replacing faulty sensor data samples with synthetic values (data imputation), which assumes that the sensor failure has been already identified [9]. For example, in case of a multivariate system, a simple nearest-neighbor estimator can be used to deduce the value of a missing sample by using data samples from other operational sensor units [9].

Another possible problem involves the network bandwidth and the energy requirements. Having multiple sensors over a large area means that it is often impractical to tether all of them together into the power line and a wired network or shared data acquisition (DAQ) unit. This means that the sensors may have to be battery-powered and transport data samples using wireless technologies, which can imply severe network bandwidth and energy restrictions. This is not necessarily a problem when the sensor units operate at a relatively low sampling rate; some temperature-based idler fault detection systems can indeed operate purely with harvested energy [11]. However, such systems have relatively low sensitivity, and high-sensitivity detectors based on vibration/acoustic emission sensors and ML classifiers are generally much more energy intensive. Furthermore, for joint classification, measurements from multiple sensor units need to be transmitted into a central data collection unit prior to classification. As deep learning-based sensing systems may need access to the raw unprocessed (or minimally processed) streams of sensor readings for feature extraction [12], the bandwidth and the energy requirements associated with data transmission can grow to an impractical level, as each sensor unit could generate and transmit up to a few megabits of data samples per second (as demonstrated in later sections).

Therefore, naively transmitting every collected data sample can quickly become impractical in the case of deep learning-based multivariate sensing systems, and data reduction techniques may be needed in order to keep the network bandwidth and energy budgets at a manageable level. Some systems use duty-cycled sensor data to reduce the energy requirements [13,14]; in these articles, the authors try to balance the system accuracy and energy efficiency of the system by carefully adjusting the duty cycle (on–off periods) of the involved sensor units. Others have used compressive sensing techniques such as sub-Nyquist sampling [15,16] or edge computing techniques such as autoencoders to compress the data [17]. These approaches try to reduce the amount of data generated by the sensor units while minimizing the loss of information by either adjusting the sampling rates or by applying lossy compression to the data. Another approach to the problem is to not install sensor units directly onto the belt conveyor, but to install them on robots or unmanned vehicles which scan the belt conveyor physically in prescheduled intervals [11]. However, such systems come with challenges of their own; for instance, because the sensor units are not directly attached to the belt, and the vehicles themselves generate vibrational and acoustic noises, the quality of the data can be degraded.

In this article, we implement and evaluate a belt conveyor idler monitoring system that detects and identifies stalled idlers by using accelerometers and microphones as cheaper substitutes for piezoelectric transducers. We analyze the reliability, network bandwidth, and energy requirements of the proposed system to evaluate the practicality of the design. Wired accelerometers and microphones installed on our belt conveyor testbed are used to collect near-ideal-condition datasets. The data samples are then intentionally modified to simulate sensor failures and measure the effect of the accelerometer/microphone sampling rates on the accuracy of the model. The results of our analysis suggest that such systems can indeed detect and locate stalled idlers accurately; however, careful consideration should be given to network and energy efficiency so as not to exceed the network and energy budgets.

The main contribution point of this article is the scalability analysis results, which address how well a deep learning-based idler fault detection model can scale in terms of network bandwidth and energy costs as well as the system's survivability when exposed to harsh operating conditions under which some sensor units can suffer failures.

The rest of this paper is organized as follows: in Section 2, we discuss the methods and equipment used to collect the dataset and the models used to implement the idler stall detection and identification system; in Section 3, we display the experimental results of the proposed system, including the model accuracy, network bandwidth, and estimated energy requirements; in Section 4, we analyze and discuss the obtained results to derive conclusions; finally, we summarize our results and wrap up the article in Section 5.

2. Materials and Methods

2.1. Data Collection

We collected datasets from our own belt conveyor testbed, consisting of a sloped v-trough belt conveyor with an upper belt length of 7 m. Analog accelerometers and microphones were installed on the frame of the belt conveyor to collect data samples, as shown in Figure 2. To simulate stalled idlers, the idlers were intentionally jammed with stones (Figure 1). When jammed in this way, the idlers were unable to rotate freely and dragged against the belt.

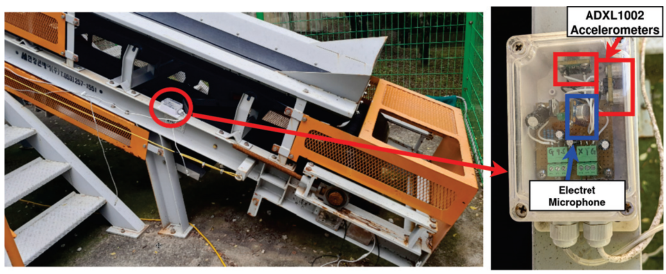


Figure 2. Design of the sensor units and their installed locations.

2.1.1. Hardware Setup

Vibration and acoustic data samples were collected using ADXL 1002 Micro-Electro-Mechanical Systems (MEMS) analog single-axis accelerometers (from Analog Devices, Wilmington, MA, USA), and electret microphones with Maxim Integrated MAX4466 amplifier modules (by Adafruit, New York, NY, USA) by connecting them to a NI-6353 DAQ unit, from National Instruments (Austin, TX, USA) with generic shielded cables. MEMS accelerometers and electret microphones were used as substitutes for piezoelectric accelerometers and acoustic emissions sensors, which are often preferred in academic research projects; the rationale behind this choice was that these types of sensors are more likely to be used in real-world deployments. While piezoelectric transducers usually provide higher sensitivity and larger bandwidth than their MEMS counterparts, they also come at much higher cost, which is difficult to justify for large-scale deployments involving hundreds to thousands of sensor units. As shown in the next section, the data quality of these MEMS accelerometers and electret microphones is mostly sufficient for our purposes. All equipments and sensor modules were imported to Pohang, South Korea via local electrical component suppliers.

Two accelerometers and one microphone were grouped to form a 'sensor unit'. Each 'sensor unit' was then enclosed in a weatherproof enclosure for outdoor installations, as shown in Figure 2. The accelerometers were pointed perpendicular to the frame, one pointing upward and the other pointing to the left, forming two orthogonal measurement axes. Because the microphone is an omnidirectional unit, it receives sound waves from every direction. In total, ten sensor units were produced and installed, providing thirty analog measurement channels in total (twenty accelerometers and ten microphones). The sensor outputs were directly sampled by a 32-channel 16-bit DAQ NI-6353 at a sampling rate of 31.25 kSa/s, which is the maximum sampling rate achievable by the DAQ when all 32 channels are scanned simultaneously. This is reasonably close to the Nyquist rate of the sensors [18,19], which is 22 kSa/s and 40 kSa/s for the accelerometers and microphones, respectively. All 32 channels of the DAQ were scanned, with 30 channels actually connected to the sensors and the two remaining unused channels connected to dummy resistors to suppress ghosting effects caused by the voltage range differences between the microphones and accelerometers [20]. The collected samples were stored in files using SigMF format version 1.0 [21], then transmitted to a centralized data server for further analysis. Python 3.11.2 and the nidaqmx Python library (version 0.9.0) were used to implement the collection software.

The locations of the sensor units are shown in Figure 3. The separation between the sensor units along the frame were approximately 1.2 m, with some variations in placement due to various physical restrictions.

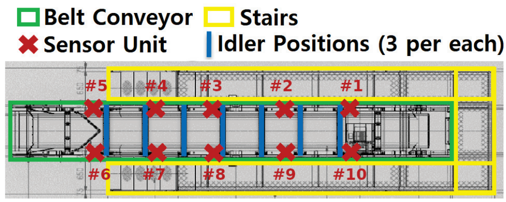


Figure 3. Locations of the sensor units and idlers. Each sensor unit contains three sensors (two accelerometers and one microphone), while each idler location contains three idlers (left, middle, and right idler). The numbers (#1–#10) denote the identification numbers of the sensor units used internally.

The overall system architectures are shown in Figure 4. It must be noted that the testbed that we used to collect the datasets and train and evaluate the models was structured differently from the expected deployment scenarios, especially in that the testbed relies on wired sensor units instead of wireless units. This was done intentionally in order to collect high-quality samples without duty cycling. The data samples were intentionally unsynchronized later on during the training and evaluation phases to simulate unsynchronized wireless sensor units.

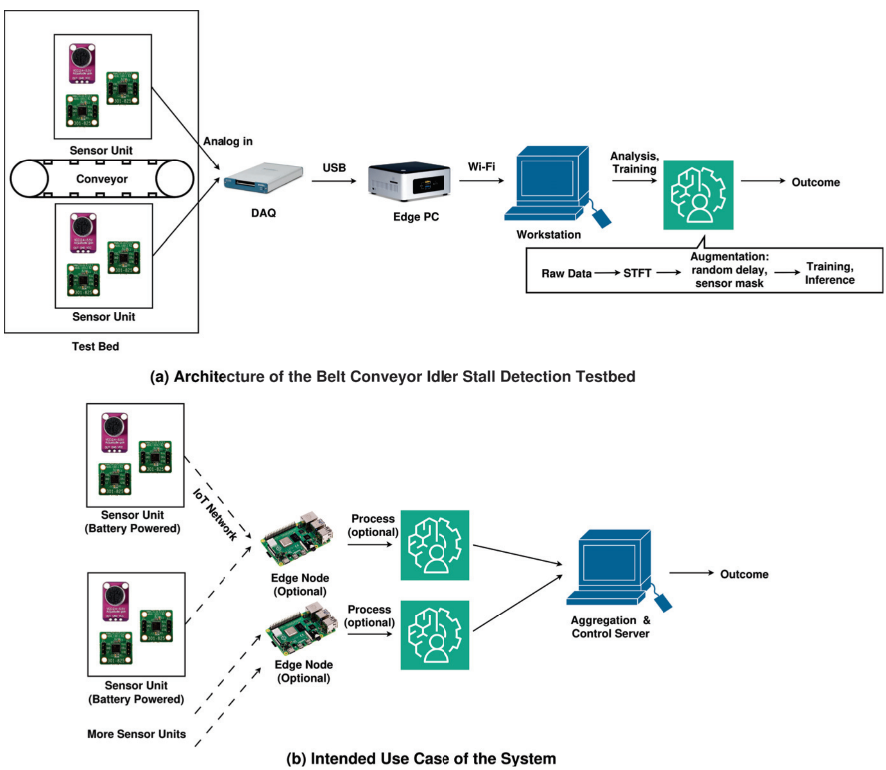


Figure 4. Overview of the system: (**a**) system architecture of the testbed used to collect the data samples and (**b**) expected system architecture in actual deployment.

2.1.2. Dataset Collection

Two types of data samples were collected: normal samples, which formed the normal dataset, and samples with a stalled idler, which formed the anomaly dataset. The normal dataset contained signal samples collected during the normal operation of the belt conveyor with no stalled idlers, while the anomaly dataset contained signal samples with one stalled idler at a known position. As shown in Figure 1, the idler was intentionally jammed to cause a stall. Due to physical constraints, only the upper left/right idlers were jammed (seven left idlers and seven right idlers, for a total of fourteen). The belt was not loaded with payloads during data collection.

The list of collected datasets is summarized in Table 1. The speed of the belt is directly related to the inverter frequency, with the slowest at 10 Hz (approximately 0.11 m/s) and the fastest at 60 Hz (approximately 0.63 m/s).

Table 1. Collected datasets for the belt idler stall detection experiments.

Condition	**Inverter Frequency (Hz)**						**Sum**
	10	**20**	**30**	**40**	**50**	**60**	
Normal	11 h	11 h	11 h	11 h	11 h	11 h	66 h
Stalled Idler	20 min × 14 idlers	20 min × 14 idlers	20 min × 14 idlers	20 min × 14 idlers	20 min × 14 idlers	20 min × 14 idlers	28 h

2.2. Model Implementation

2.2.1. Design Considerations

State-of-the-art idler fault detection systems usually rely on ML algorithms to process sensor data and to identify faults [2]. While the properties of one of the most common failure modes, namely, bearing faults, have been closely analyzed and can be inspected algorithmically [22], this is not the only possible failure mode, and the observed power spectra can be vastly different even in the case of bearing faults. This is because the vibrations and acoustic noises created by the belt conveyor are not solely from the bearings, but from various interactions; for example, the drag induced by the stalled bearing can also generate various frequency components in the accelerometer and microphone sensor data which cannot be easily accounted for.

Various neural network-based ML models have been considered, including simple multilayer perceptrons, 1D convolutional neural networks (CNN), recurrent neural networks, and 2D-CNNs. In addition, various preprocessing techniques for the samples have been considered, including the discrete Fourier transform (DFT) for ML models requiring 1D data shapes and the short-time Fourier transform (STFT) and continuous wavelet transform for ML models requiring image-like 2D input data shapes. We selected a 2D-CNN, as it showed the most promising results during our preliminary analysis.

It must be noted that STFT-based models tend to require more sample points to make an inference than the other alternatives considered above. This is because the number of sample points needed to calculate an STFT matrix is equal to twice the number of the elements in the matrix when no overlaps are assumed. Wavelet transform-based models and multivariate 1D time series-based models tend to require fewer sample points to make an inference, although this is implementation-dependent and not necessarily always the case. Nonetheless, this means that our STFT-based model is likely to suffer from higher network overhead than alternative designs. While this is not favorable from the viewpoint of scalability, proving the practicality of the model with STFT would also demonstrate the practicality of the system with smaller models that would require less network bandwidth to operate.

2.2.2. Model Design

Multivariate time series sensor data streams are converted into time-frequency matrices with STFT, then inputted to a 2D-CNN. MobileNet V2 [23] was modified to allow 30-channel STFT input data for use as our classification model. This model detects whether an idler stall has occurred; in the event of a fault, it also identifies the location of the stalled idler. The output of the model (the expected label) is either 0 if no idler is stalled, or the position of the stalled idler (1–14) if a stall is detected. Python 3.11.2 and PyTorch 2.0.1+cu117 were used to implement the model.

2.2.3. Data Augmentation for Sensor Fault Tolerance

Augmented samples are used to train the model. The channels of the input data were randomly selected and masked with zeros to simulate sensors that are missing due to sensor malfunctions or communication failures caused by network instability. A discrete uniform distribution with a probability of 0.5 was used to select and drop the channels (sensors). In this way, the trained model learns to deal with missing sensor data.

We assumed that failed sensor units transmitted either no data or data consisting of only zeros. While this is not necessarily true, similar effects can be achieved by implementing a sensor failure detection system [8] and filtering out any faulty samples detected with it.

Every random parameter used in the augmentation process was recalculated every time the augmentation was applied in order to maximize the randomness of the results.

2.2.4. Preprocessing

The augmented samples were converted into time–frequency domain matrices by applying STFT. The elements in the STFT matrices were then squared to obtain power estimates, converted to the logarithmic scale (base-10), and normalized to approximately [0, 1] using pre-estimated normalization factors. The resulting matrices were used as the input features of the CNN model.

2.2.5. Training and Evaluation

The implemented model was trained using the collected data samples split in an 8:1:1 training–validation–testing ratio. Supervised learning was used to train the model. The hyperparameters and various configurations used to train the model are displayed in Table 2.

Samples with an inverter frequency of 30 Hz were intentionally excluded in order to evaluate the generalization performance of the model. All other samples were used as either training samples, validation samples, or testing samples. The excluded 30 Hz samples were used later to test the accuracy of the model on unseen environments (unseen inverter configurations), and are referred to as the "unseen test data".

Table 2. Default hyperparameters and configurations used to train the proposed idler stall identification model.

Name	Value
Input Shape	$30 \times 256 \times 256$
Preprocessing Methods	STFT
Augmentation	channel mask
Batch Size	16
Learning Rate	0.0005
Epochs	50 (with early stopping)
Dataset Split	8:1:1 (train:val:test)
Loss Function	Cross Entropy loss
Optimizer	Adam

In addition to the main identification model, an additional model was trained with the augmentation step disabled in order to evaluate the effectiveness of the augmentation in hardening the model. In addition, several more models were trained with decimated samples to evaluate the minimum acceptable sampling rates of the system. Two different downsampler configurations were considered: the first with an ideal (DFT-based) low-pass filter, and the second without the low-pass filter, potentially allowing high-frequency components to be aliased into low-frequency regions.

3. Results

3.1. Network Bandwidth and Energy Requirements

The network and energy usages of the sensor units were calculated to evaluate the scalability of the system.

3.1.1. Theoretical Limits

The theoretical bound of the network bandwidth and energy requirements (of the communication circuits) can be estimated using the Shannon capacity and a channel model.

The Shannon capacity of a wireless channel is

$$C = B \log_2(1 + \frac{S}{N}), \tag{1}$$

where C is the channel capacity (bits/s), B is the bandwidth of the channel (in Hz), and S/N is the signal-to-noise ratio of the received signal. The available bandwidth is usually controlled by government agencies (the FCC in the case of the U.S.), and is not under our control. The noise power N is higher than or equal to the Johnson–Nyquist noise, which is approximately -174 dBm/Hz (linear-log scale conversion may be needed). Finally, the signal power S is approximated by dividing the power loss by the transmission power (assuming a linear scale), which is environment-dependent. If free (open) space is assumed, then the free space path loss equation can be used to estimate the loss:

$$\frac{P_r}{P_t} = D_t D_r (\frac{\lambda}{4\pi d})^2 \tag{2}$$

where P_r is the received power (linear scale), P_t is the transmitted power, D_t is the transmitter antenna gain, D_r is the receiver antenna gain, λ is the signal wavelength, and d is the distance between the transmission and reception antennas.

By plugging this in to the Shannon capacity and integrating the capacity equation, it is possible to estimate the minimum energy required to transmit the data.

However, the estimated results can be significantly off from the real-world results, as the exact path loss model and various energy overheads experienced by the transceivers are not known. Thus, instead of using the theoretical bounds, we simply use empirical efficiency figures advertised by Wi-Fi HaLow [24] in the rest of this article. Wi-Fi HaLow is a variation of the Wi-Fi standard optimized for long-distance machine-to-machine communication systems. The claimed energy efficiency of a typical Wi-Fi HaLow node is 22.4 kbits/J [24], that is, 22.4 kbits of data transmission per 1 joule of energy.

The physical layer of Wi-Fi HaLow is based on a modified IEEE 802.11ac [25] physical layer (PHY), downclocked to 10% of the original speed of the IEEE 802.11ac; therefore, the theoretical maximum throughput of a Wi-Fi HaLow PHY is approximately 86.7 Mbps, which is one-tenth the maximum throughput of IEEE 802.11ac. The realistic maximum throughput is usually much lower, probably somewhere around 10 Mbps depending on various environmental factors.

3.1.2. Continuous Data Streaming

The minimum network bandwidth required by our system to stream every sample into the centralized data collector server continuously without downsampling or duty-cycling is as follows:

$$31,250 \text{ Sa/s} \times 16 \text{ bits/Sa} \times 30 \text{ channels} = 15 \text{ mbps}. \tag{3}$$

The required network bandwidth per sensing channel to transmit every sample would then be

$$31,250 \text{ Sa/s} \times 16 \text{ bits/Sa} = 0.5 \text{ mbps}. \tag{4}$$

In our case, the system managed to stream every sample to our centralized server over Wi-Fi in real time without issue; however, such luxuries cannot be expected in more realistic large-scale belt conveyor installations. For example, as discussed above, the 15 Mbps bandwidth requirement is fairly close to the achievable limit of a realistic Wi-Fi HaLow network. A typical 5G New Radio (NR) Enhanced Mobile Broadband backhaul can support approximately 100 Mbps of uplink speed [26], which is lower than the downlink speed due to the asymmetricity of the uplink and downlink connections. This means that a high-performance 5G modem should be able to handle approximately 200 sensor channels (accelerometers or microphones) in our setups without applying data reduction techniques. Adding additional transceivers may not necessarily improve the situation, as the spatial diversity can quickly become the limiting factor. In such cases, data reduction (duty cycling, downsampling, compression, etc.) is unavoidable. Other IoT-optimized wireless communication protocols such as ZigBee and LoRa have very limited uplink bandwidth and do not meet our bandwidth requirements, as these protocols are generally not designed to carry raw data samples directly.

3.1.3. Per-Inference Network Requirements

To create the 256 × 256 × 30 spectrogram used by our inference model, we need to transmit

$$256 \times 256 \times 30 \times 2 \times 2 \text{ bytes/Sa} = 7.5 \text{ MiB} \tag{5}$$

of raw data to the central processing server. Similarly, the per-channel cost of the system is 0.25 MiB/ch. Using the 22.4 kbits/joule energy efficiency figure from Wi-Fi HaLow [24], the required amount of energy can be estimated as follows:

$$\frac{7.5 \text{ MiB}}{22.4 \text{ kbits/J}} = 2740 \text{ J} = 0.76 \text{ Wh}, \tag{6}$$

amounting to approximately 91.4 J (0.025 Wh) per sensing channel, or 0.076 Wh for each sensor unit with three sensing channels.

3.2. Data Visualization

The power spectral density (PSD) estimates of the normal dataset and anomaly dataset (idler #1 stall, which is the nearest idler to the sensor unit #1) at the inverter frequencies of 10 Hz and 60 Hz are shown in Figures 5 and 6, providing a visualization of the collected data. In addition, the spectrograms of the first sensor unit during normal operation and during the idler #1 stall are shown in Figure 7. While there are some minor differences in the plots, distinguishing the normal data from the anomaly data is difficult when using only the PSD estimates. The STFT spectrograms are easier to distinguish, as the spectrograms from the anomalous samples contain intermittent broadband frequency components.

3.3. Model Performance

The classification accuracy of the model is shown in Tables 3 and 4, along with the effect of data augmentation (channel masking). The classification accuracy of the model is reasonably high even when exposed to an unseen domain (inverter frequency of 30 Hz).

3.3.1. Effects of Data Augmentation and Sensor Failures

The classification accuracy results of the models trained with and without channel mask augmentation are displayed and compared in Tables 3 and 4. Three different configurations were evaluated: with every sensor channel available, with 20% of the sensor channels masked by zeros, and with 80% of the sensor channels masked by zeros.

The results displayed in Table 3 were calculated using test samples from the observed environments (inverter frequencies of 10–20 Hz and 40–60 Hz), while the results in Table 4 were calculated with samples from the unseen environment (inverter frequency of 30 Hz), which were excluded from the training dataset. The results on the unseen environment show that resiliency of the model trained with augmentation is vastly better than the model trained without augmentation.

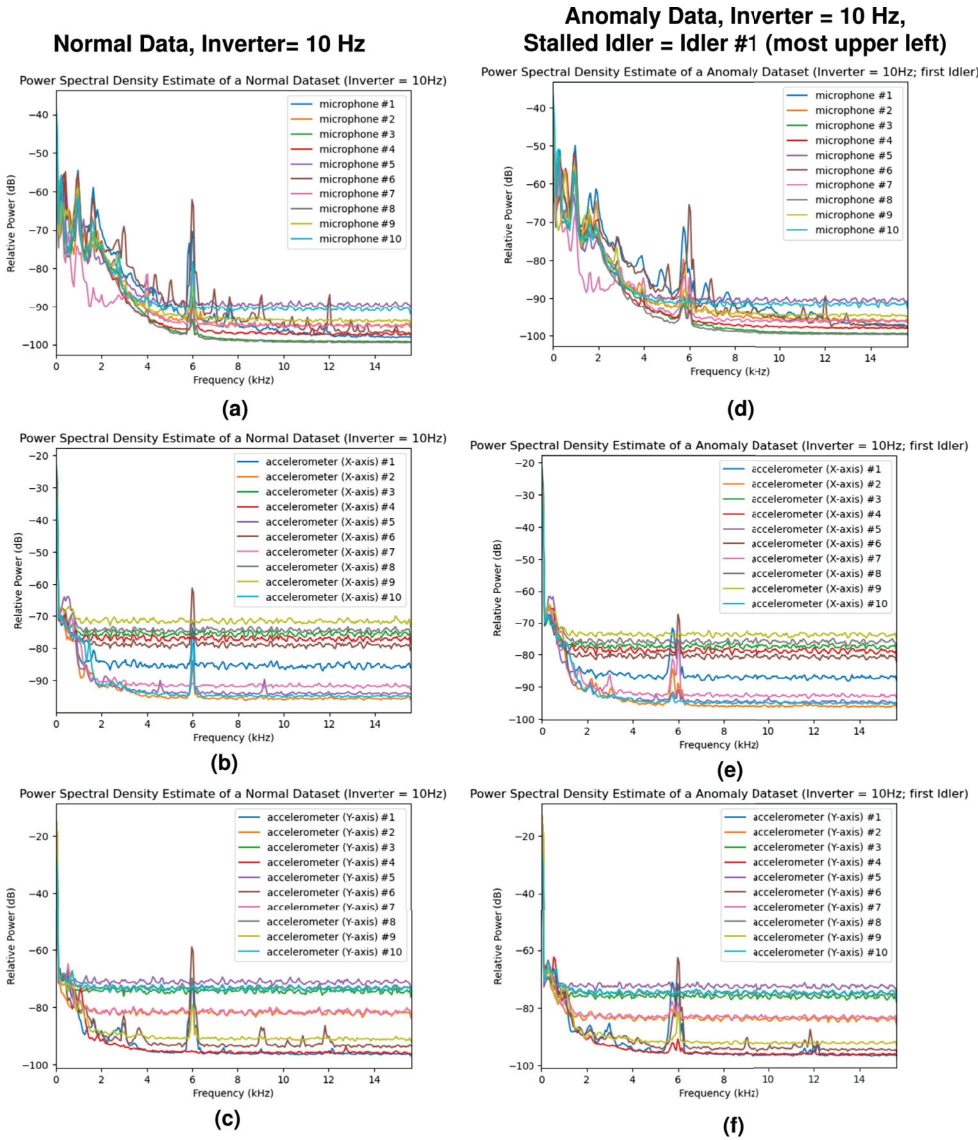


Figure 5. Power spectral density (PSD) estimates of the sensor data for an inverter frequency of 10 Hz: (**a**,**d**) PSD estimates of the microphone sensors for the normal and anomaly datasets, (**b**,**e**) PSD estimates of the x-axis accelerometer sensors, and (**c**,**f**) PSD estimates of the y-axis accelerometer sensors. The sensor numbers (#1–#10) in the plots above correspond to the sensor identification numbers in Figure 3.

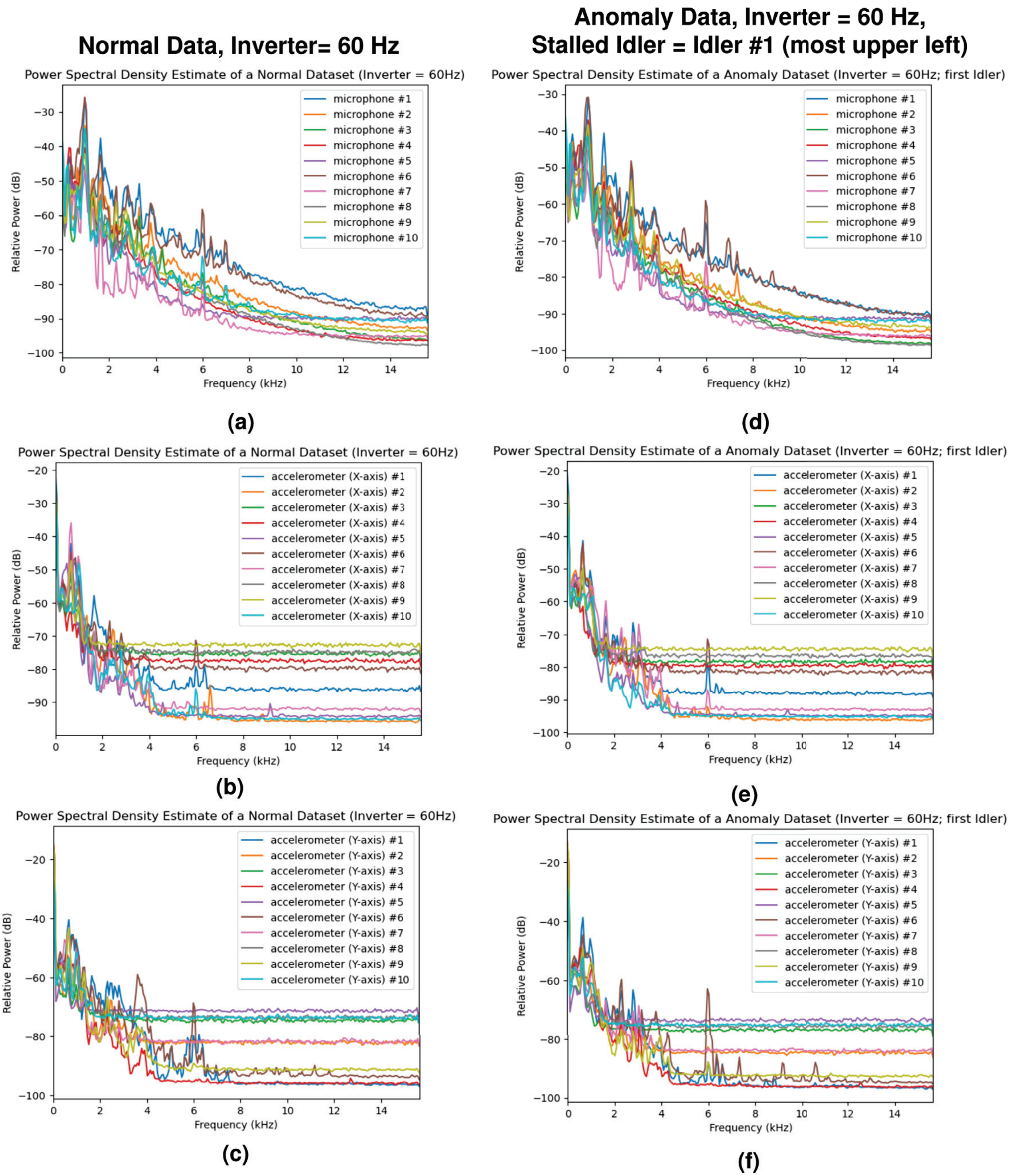


Figure 6. Power spectral density (PSD) estimates of the sensor data for an inverter frequency of 60 Hz: (**a**,**d**) PSD estimates of the microphone sensors for the normal and anomaly datasets, (**b**,**e**) PSD estimates of the x-axis accelerometer sensors, and (**c**,**f**) PSD estimates of the y-axis accelerometer sensors. The sensor numbers (#1–#10) in the plots above correspond to the sensor identification numbers in Figure 3.

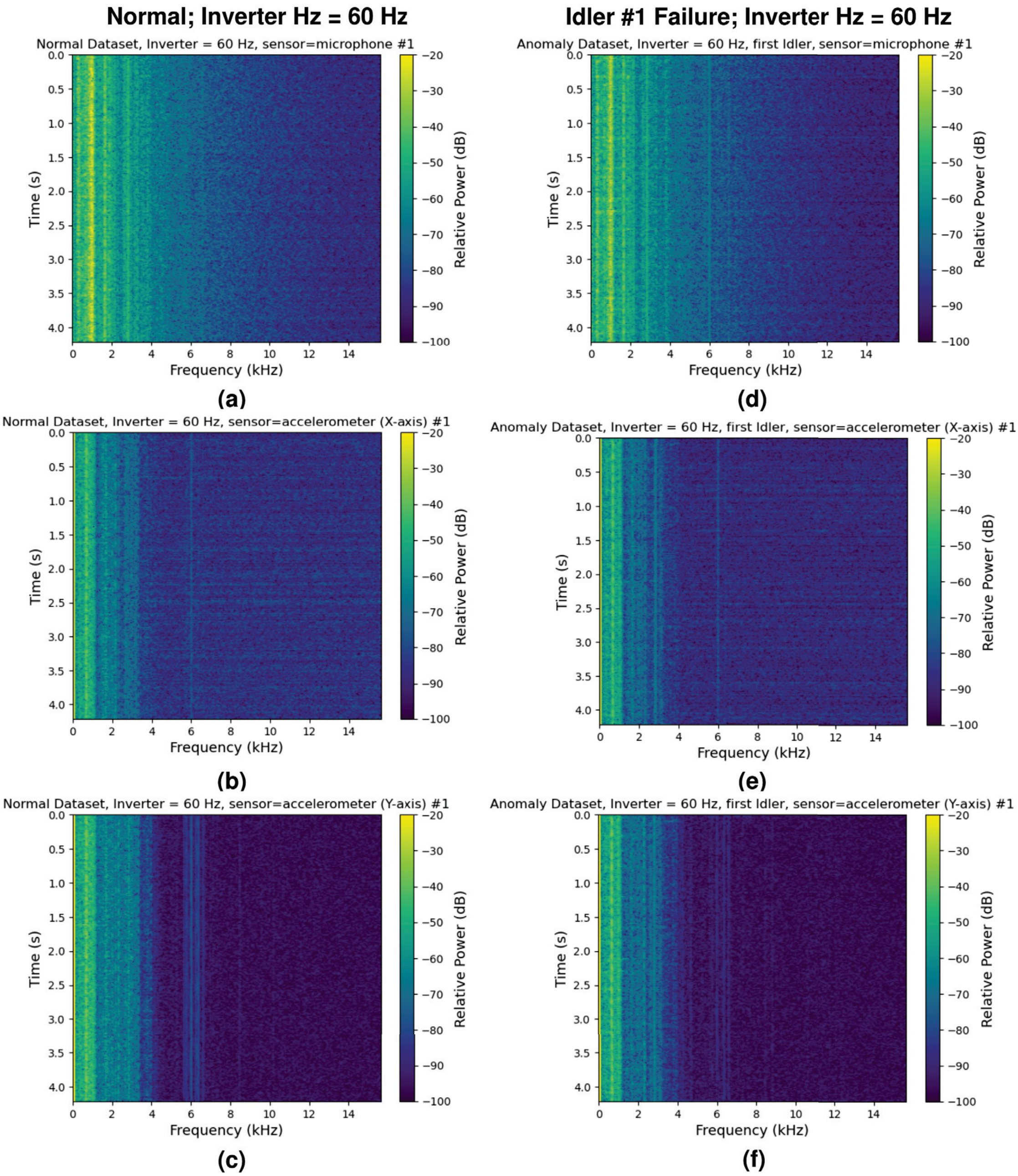


Figure 7. Spectrograms of the first sensor unit in the case of the normal operation (**a**–**c**) and in case where the nearest idler is stalled (**d**–**f**) for an inverter frequency of 60 Hz; (**a**,**d**) spectrograms of the microphone, (**b**,**e**) spectrograms of the *x*-axis accelerometer sensor, and (**c**,**f**) spectrograms of the *y*-axis accelerometer sensor. The sensor numbers (#1–#10) in the plots above correspond to the sensor identification numbers in Figure 3.

Table 3. Model accuracy evaluation results with samples from observed environments (inverter frequencies of 10–20 Hz and 40–60 Hz).

Channel Mask (% Masked)	Model Training Configurations	
	Without Augmentation	With Augmentation
No Mask	99.522%	99.952%
10% masked	7.321%	99.856%
20% masked	6.770%	99.450%
30% masked	7.201%	99.474%
40% masked	4.833%	99.593%
50% masked	3.469%	99.761%
60% masked	3.469%	98.876%
70% masked	3.469%	97.584%
80% masked	3.469%	91.340%
90% masked	3.469%	58.158%

Table 4. Model accuracy evaluation results with samples from the unseen environment (inverter frequency of 30 Hz).

Channel Mask (% Masked)	Model Training Configurations	
	Without Augmentation	With Augmentation
No Mask	96.339%	97.100%
10% masked	7.237%	99.783%
20% masked	10.330%	96.653%
30% masked	4.869%	95.360%
40% masked	3.467%	88.740%
50% masked	3.443%	88.232%
60% masked	3.443%	83.279%
70% masked	3.443%	83.569%
80% masked	3.443%	75.933%
90% masked	3.443%	49.305%

3.3.2. Effects of the Input Data Dimensions

Next, we evaluated the effects of the input data dimensions on the accuracy of the model. The choice of input dimensions can affect the accuracy of the model along with the network bandwidth, energy requirements, and/or classification intervals.

The results in the Table 5 were obtained by progressively reducing the number of the time series accelerometer and microphone samples used to detect and identify faults. The DFT size and the overlap length of the STFT operations were left unmodified. It can be seen that the accuracy starts to drop quickly when the input dimensions are reduced below $30 \times 64 \times 256$.

Table 5. Model accuracy evaluation results with varying input data dimensions.

Input Dimensions (ch × len × DFT_dim)	Accuracy (Seen)	Accuracy (Unseen; 30 Hz)	Transmitted Data (Per-Sensor)	Transmission Energy (Calculated, Per-Sensor)
$30 \times 512 \times 256$	100.0%	96.537%	1.5 MiB	0.152 Wh
$30 \times 256 \times 256$	100.0%	96.230%	0.75 MiB	0.0762 Wh
$30 \times 128 \times 256$	99.95%	94.177%	0.375 MiB	0.0381 Wh
$30 \times 64 \times 256$	99.36%	83.467%	0.188 MiB	0.0190 Wh
$30 \times 32 \times 256$	88.665%	68.961%	0.094 MiB	0.0095 Wh

3.3.3. Effects of Decimation and Downsampling

This subsection evaluates the effects of downsampling the data samples. This knowledge can allow us to better understand the sampling rates and analog bandwidth requirements of the sensor units, which can be used to optimize the network bandwidth requirements of the system and permit the use of inexpensive lower-bandwidth sensor hardware.

Two different downsampling configurations were evaluated, namely, downsampling with and without a low-pass filter. In the former case, an ideal integer downsampler with an ideal (square) low-pass filter was simulated by calculating spectrograms and dropping the high-frequency components from the calculated spectrograms. The resulting spectrograms were then directly input to the CNN models for identification. In the latter case, the raw samples were directly decimated without applying a filter. This causes high-frequency components in the samples to be down-aliased. While aliasing is often considered to be bad, it can actually help the system to retain its classification accuracy when the spectrum is sparse by behaving as a primitive compressive sensing system.

In the evaluation results shown in Tables 6–9, the first two tables display the performance of the systems when assuming that every sensor unit is available, while the later two show the performance when assuming that only 20% of the sensor units are available. The models were retrained every time the downsampling configurations were changed, as the downsampling process can remove or shift (by aliasing) the frequency components that are utilized by the model for the classifications.

Table 6. Model accuracy evaluation results with downsampled accelerometer/microphone samples (no low-pass filter applied).

Downsample Factor	Input Dimensions (ch × len × DFT)	Accuracy (Seen)	Accuracy (Unseen; 30 Hz)	Transmitted Data (Per-Sensor)
×1	30 × 256 × 256	99.904%	97.100%	0.75 MiB
×2	30 × 256 × 128	99.833%	91.929%	0.375 MiB
×4	30 × 256 × 64	99.904%	82.711%	0.188 MiB
×8	30 × 256 × 32	99.856%	91.253%	0.094 MiB
×16	30 × 256 × 16	96.938%	91.966%	0.047 MiB
×32	30 × 256 × 8	96.675%	76.392%	0.023 MiB
×32	30 × 32 × 64	99.737%	79.352%	0.023 MiB

Table 7. Model accuracy evaluation results with downsampled accelerometer/microphone samples when applying an ideal DFT low-pass filter.

Downsample Factor	Input Dimensions (ch × len × DFT_dim)	Accuracy (Seen)	Accuracy (Unseen; 30 Hz)
×1	30 × 256 × 256	99.952%	97.100%
×2	30 × 256 × 128	99.904%	91.699%
×4	30 × 256 × 64	99.330%	88.208%
×8	30 × 256 × 32	97.464%	97.378%
×16	30 × 256 × 16	98.780%	81.624%
×32	30 × 256 × 8	94.952%	83.835%
×32	30 × 32 × 64	97.967%	76.754%

There are some fluctuations in the evaluation results; the changes in the accuracy are not monotonic when the downsampling factors are increased/decreased, which would have been the case if the system and the models were ideal. This is probably because the CNN models were retrained each time configuration changes were made. The models may have ended up in different local minima due to the randomness involved in the training process. Nonetheless, the overall trends of the results agree with our expectations. The system can definitely cope with somewhat reduced sampling rates, though it is difficult to draw a clear line due to the aforementioned fluctuations.

Table 8. Model accuracy evaluation results using downsampled accelerometer/microphone samples (no low-pass filter applied) with eight out of ten sensor units disabled (masked).

Downsample Factor	Input Dimensions (ch × len × DFT_dim)	Accuracy (Seen)	Accuracy (Unseen; 30 Hz)
×1	30 × 256 × 256	77.751%	67.645%
×2	30 × 256 × 128	81.746%	64.589%
×4	30 × 256 × 64	81.220%	74.931%
×8	30 × 256 × 32	70.454%	52.906%
×16	30 × 256 × 16	71.770%	80.766%
×32	30 × 256 × 8	62.249%	66.689%
×32	30 × 32 × 64	65.646%	51.371%

Table 9. Model accuracy evaluation results, using downsampled accelerometer/microphone samples (ideal DFT low-pass filter applied) with eight of ten sensor units disabled (masked).

Downsample Factor	Input Dimensions (ch × len × DFT_dim)	Accuracy (Seen)	Accuracy (Unseen; 30 Hz)
×1	30 × 256 × 256	77.727%	67.657%
×2	30 × 256 × 128	95.789%	63.115%
×4	30 × 256 × 64	73.493%	62.305%
×8	30 × 256 × 32	67.201%	66.075%
×16	30 × 256 × 16	69.019%	62.921%
×32	30 × 256 × 8	67.727%	54.633%
×32	30 × 32 × 64	80.167%	49.583%

4. Discussion

As shown in Table 3, the accuracy of the implemented model is very high; however, it is not yet ready for real-world deployment. As discussed previously, the scalability of the system still needs to be addressed. In this section, we discuss possible change to the model based on the experimental results presented in the previous section.

4.1. Comparison with Previous Works

It is difficult to compare our proposed model against previous works in an objective manner due to the differences in testbed configurations and sensor configurations. In particular, most previous systems have been designed and tested using only a single sensor unit, i.e., a univariate sensing system, which is not suitable for the sensor configurations used in our testbed.

Nonetheless, comparing these prior works with our system can provide invaluable insights about the proposed system. Therefore, we describe the results from previous studies in Table 10 below. Please note that the performance figures in the table below were obtained from the original papers and were not recalculated or re-evaluated with our own datasets, and as such cannot be compared directly.

We have used a relatively simple feature extractor and classifier model compared to other researchers; on the other hand, our proposed system probably has an edge in terms of the quality of the data samples. We have mainly concentrated on the scalability and data quality of the proposed system, while previous researchers placed more effort on optimizing the classification models.

4.2. Scalability: Operating with a Subset of Sensors

In Tables 3 and 4, we show that the resiliency of the model against missing sensor streams can be vastly improved by simply augmenting the training data. While the main rationale behind this was to harden the model against sensor failures, it can be also used to save network bandwidth and energy consumption by allowing the model to operate with only a subset of the sensor units.

Table 10. Comparison of the proposed model against previous works. Note that the accuracy measurements have not been recalculated using our datasets.

Parameter	Ours	Alharbi, F. [5]	Yang, M. [6]
Year	2024	2024	2020
Sensor Type	Accel + Microphones (multivariate)	Microphones	Microphones
Feature Extraction	STFT	BPF, Neural Networks (YAMNet, BiLSTM)	varies (handcrafted, autoencoder, etc.)
Classifier	CNN (MobileNet V2)	XGBoost	varies (SVM, DNN, CNN attempted)
Sample length	4.2 s	0.48 s	14 s
Accuracy [1]	99.9% (best case)	approx. 95% [2]	96.7%

[1] Results as reported in the original paper. [2] The goal of the classifier is slightly different from ours (failure stage estimation).

By only using a subset of sensor units for inference, the proposed system can save a significant amount of energy and network bandwidth. The results in Table 3 suggest that only four or five out of the ten sensor units are actually needed to make an accurate classification in the case of our configuration. This can be implemented by simply turning off the sensor units and replacing the missing data samples with zeros.

Further generalizations of the model in this aspect could make for an interesting future research topic; for example, having a model that can dynamically select sensor units for inference could make the maintenance and sectorization processes much more flexible. We speculate that it would be possible to implement a reinforcement learning-based sensor controller logic which dynamically connects/disconnects the sensor units as needed to minimize the accuracy loss while maximizing the maintenance interval by conserving the battery to maximize a target utility function. A similar concept has already been proposed and demonstrated, with the authors demonstrating that reinforcement learning can be used to select the best subset of temperature sensor units for monitoring the temperature of a given area [27].

4.3. Scalability: Bandwidth and Energy Considerations

As shown in the previous section, the amount of data needing to be transmitted by our idler stall identification system is 7.5 MiB per inference (0.75 MiB per sensor unit) when no data reduction techniques are applied. Assuming a Wi-Fi Halow-based network backhaul, this means that approximately 0.76 Wh of energy is used per transmission (0.076 Wh per sensor unit).

The energy costs of the other parts of the sensor hardware are usually much lower, and consequently mostly negligible: the ADXL1002 accelerometer needs approximately 0.005 W [18], the microphone module uses approximately 0.05 W [19,28], and the ADC and MCU boards use approximately 0.038 W and 0.1 W of power, respectively (assuming an ADS1256 ADC and STM32F0512 MCU) [29,30]. Therefore, the combined power draw of the sensor unit, excluding the wireless components, is approximately 200 mW (pessimistically). This means that the sensor unit uses 0.2 Wh of energy per hour, assuming that every chip is always in full operation. Meanwhile, the energy required for a sample transmissions is approximately 0.076 Wh per inference, which can occur up to tens of times per minute if no duty cycling is used.

If a sensor unit has a 3000 mAh 3.7 V Li-ion battery installed, a brand-new battery would last for

$$\frac{3\text{ Ah} \times 3.7\text{ V}}{0.076\text{ Wh}} = 145\text{ inferences} \tag{7}$$

counting only the energy used for wireless data transfer. This means that without mainline power or a large solar panel, the battery of the sensor unit may quickly drain if no data reduction is applied.

The simplest method to reduce this energy cost is to duty-cycle the sensors, that is, to slow down the inference intervals and put the sensor units into sleep mode whenever inference is not in progress. While this is one of the most commonly used power-saving methods [11], it comes at the cost of slower identification speed and slightly reduced sensitivity of the system due to the reduced number of inferences made in a unit of time. Thus, while duty cycling can increase the battery lifespan and scalability of the system to almost an arbitrary scale, its use must be minimized whenever possible.

Fortunately, there are many other opportunities to reduce the bandwidth requirements of the system. For example, the results in Sections 3.3.2 and 3.3.3 (Tables 6–9) suggest that the sampling rates and length of the input data of our system are somewhat redundant. Because the network bandwidth and energy requirements are directly proportional to these parameters, simply cutting the sampling rates to a quarter and the sample lengths to half would reduce the associated costs by one-eighth. Even further reductions would be possible with some loss of accuracy. When paired with a high-throughput network backhaul (such as 5G NR), the one-eighth reduction rate would allow the system to support approximately 1000–2000 sensor channels concurrently.

By combining duty cycling with downsampling and downsizing techniques, the network efficiency and the battery life of the sensor units can be significantly boosted. For example, if the sensor unit were duty cycled in a such way that it would transmit one data sample per hour and the data rate reduction of one-eighth were applied to the sensor, the sensor would last approximately

$$\frac{3\text{ Ah} \times 3.7\text{ V}}{(0.076/8)\text{ Wh}} = 1168\text{ inferences,} \tag{8}$$

which is approximately 1168 h, or 48 days. With larger battery modules and/or further data reductions (1/16 reduction with subset selection, for example), it would be possible to improve the battery life to approximately a year.

One of the insights obtained from the spectrograms of our sensor data (Figure 7) is that the anomalous data samples often contained intermittent wideband frequency components (Figure 7d) which were not present in the normal samples. This could explain why the STFT-based model showed a higher accuracy over the alternative designs that we considered during our preliminary analysis. STFT can capture both the short-term power estimates and the long-term power fluctuations, which were not captured well by other preprocessing techniques we tested, such as continuous wavelet transforms. If this is indeed the case, then designing simpler handcrafted features that capture both aspects of the spectrum could allow for further reduction of the data rates. Cyclostationary analysis techniques such as those in [31] could also prove helpful if any repetitive patterns could be found in the measurements.

4.4. Long-Term Stability

One of the challenges that is unaddressed in this article and left as future work is improving the long-term stability of the system. A common problem of supervised fault detection models such as the one proposed in this article is that the accuracy of the system tends to degrade over time. This is because such models are often not designed with environmental variables in mind; as time passes, the surrounding environment often changes, and the various properties of the system, including the conveyor and the sensors themselves, may undergo drift as they age. While this would not happen in an ideal world where the model is trained with an infinite amount of data samples from various domains, real-world datasets are much more restricted and biased, causing the model to overfit to specific conditions.

One way to overcome this problem is to use handcrafted features designed to be mostly independent from external environmental factors and aging of the hardware. An alternative approach to solve this problem would be to use unsupervised or semi-supervised learning techniques, which would allow the model to be automatically retrained using more recent unlabeled samples as needed and without human intervention. One such example is an autoencoder-based unsupervised anomaly detection model consisting of an autoencoder model (a deep learning-based lossy data compression–decompression model) intentionally overfitted to normal operating conditions. Because the model is overfitted to normal conditions, it does an excellent job of compressing normal samples, but does not compress abnormal samples as efficiently. This can be used to detect the presence of anomalies [32]. The training process is mostly unsupervised, as the system (the conveyor) mostly operates in normal condition; thus, most of the collected data samples are from these normal operating conditions, and the model remains likely to overfit to them even when abnormal samples are not filtered out carefully.

Continual learning could also be a possible solution to the same problem. As the name suggests, continual learning is a technique involving continuous training and updating of the model even after it is deployed in the field. In this case, the original training datasets are usually not expected to be available. Common workarounds to the problem include generating substitute samples using generative networks or carefully configuring the regularization factors to prevent the model from experiencing catastrophic forgetting. Recently, there have been attempts to bring such techniques to the world of anomaly detection systems, for example, in [33]. Implementing such systems can help the model to retain its performance even over a long period.

5. Conclusions

Conveyor belt idlers are cylindrical supports used to support the belts of belt conveyor systems, and can cause major damage to the belts as they fail. Therefore, early detection of idler failures is crucial to maintaining the condition of belt conveyor systems. In this article, we have implemented and evaluated an idler stall detection system using our belt conveyor testbed equipped with accelerometer and microphone sensor units. The scalability of the implemented system in terms of resilience to sensor failures, network bandwidth requirements, and energy requirements is carefully analyzed. The results suggest that our proposed system can easily cope with a few sensor failures, and can be scaled up to meet the requirements of larger belt conveyor installations; however, the energy budgets of the sensor units must be carefully considered, as wireless data transmissions can quickly drain the batteries of the sensor units.

Future research topics involving the proposed system include the possibility of dynamically connecting/disconnecting the sensor units to the network to conserve battery life, as well as the possibility of long-term performance variations of the model and workarounds/fixes, if needed.

Author Contributions: Methodology: K.S.S. and Y.N.; software: K.S.S.; validation: K.S.S. and Y.N.; formal analysis: K.S.S., Y.N. and Y.-J.S.; investigation: K.S.S.; resources: K.S.S. and Y.N.; data curation: K.S.S. and Y.-J.S.; writing—original draft preparation: Y.N. and K.S.S.; writing—review and editing: K.S.S., Y.N. and Y.-J.S.; visualization: K.S.S.; supervision: Y.-J.S.; project administration: Y.N. and Y.-J.S.; funding acquisition: Y.-J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partly supported by POSCO DX (23DO1112), by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2022R1A6A1A03052954), by an Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH)), and through a Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Ministry of Trade, Industry & Energy (MOTIE) of the Republic of Korea (No. 20214810100010).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets presented in this article are not readily available due to an another ongoing study with overlaps in the dataset. Requests to access the datasets should be directed to the authors of the article.

Conflicts of Interest: The authors declare no conflicts of interest. The research was originally initiated by one of the sponsors (POSCO DX), but has largely diverged since then, and the research presented in this article shares little to no commonality with the original project except the use of the belt conveyor testbed and overlaps in some data samples. Therefore, the sponsors had no meaningful role in the design, execution, interpretation, or writing of the study.

References

1. Liu, X.; Pang, Y.; Lodewijks, G.; He, D. Experimental research on condition monitoring of belt conveyor idlers. *Measurement* **2018**, *127*, 277–282. [CrossRef]
2. Alharbi, F.; Luo, S.; Zhang, H.; Shaukat, K.; Yang, G.; Wheeler, C.A.; Chen, Z. A Brief Review of Acoustic and Vibration Signal-Based Fault Detection for Belt Conveyor Idlers Using Machine Learning Models. *Sensors* **2023**, *23*, 1902. [CrossRef] [PubMed]
3. Yang, B.Y. Fibre Optic Conveyor Monitoring System. Master Thesis, The University of Queensland, Brisbane, Austrailia, 2014.
4. Ibarra-Cabrera, M.J.; Rios, J.G.; Ovalle, D.V.; Aquino-Cruz, M.; Calderon-Vilca, H.D.; Ochoa, S.F. An IoT Solution to Detect Overheated Idler Rollers in Belt Conveyors. *Int. J. Adv. Comput. Sci. Appl.* **2024**, *15*, 297–304. [CrossRef]
5. Alharbi, F.; Luo, S.; Zhao, S.; Yang, G.; Wheeler, C.; Chen, Z. Belt Conveyor Idlers Fault Detection Using Acoustic Analysis and Deep Learning Algorithm With the YAMNet Pretrained Network. *IEEE Sens. J.* **2024**, *24*, 31379–31394. [CrossRef]
6. Yang, M.; Zhou, W.; Song, T. Audio-based fault diagnosis for belt conveyor rollers. *Neurocomputing* **2020**, *397*, 447–456. [CrossRef]
7. Morocco's Fish Fight: High Stakes over Western Sahara. Available online: https://www.bbc.co.uk/news/world-africa-16101666 (accessed on 30 October 2024).
8. Li, D.; Wang, Y.; Wang, J.; Wang, C.; Duan, Y. Recent advances in sensor fault diagnosis: A review. *Sens. Actuators A Phys.* **2020**, *309*, 111990. [CrossRef]
9. Faizin, R.N.; Riasetiawan, M.; Ashari, A. A Review of Missing Sensor Data Imputation Methods. In Proceedings of the 2019 5th International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 30–31 July 2019.
10. Du, J.; Hu, M.; Zhang, W. Missing Data Problem in the Monitoring System: A Review. *IEEE Sens. J.* **2020**, *20*, 13984–13998. [CrossRef]
11. Morales, A.S.; Aqueveque, P.; Henriquez, J.A.; Saavedra, F.; Wiechmann, E.P. A Technology Review of Idler Condition based Monitoring Systems for Critical Overland Conveyors in Open-pit Mining Applications. In Proceedings of the 2017 IEEE Industry Applications Society Annual Meeting, Cincinnati, OH, USA, 1–5 October 2017.
12. Gjoreski, H.; Bizjak, J.; Gjoreski, M.; Gams, M. Comparing Deep and Classical Machine Learning Methods for Human Activity Recognition using Wrist Accelerometer. In Proceedings of the IJCAI 2016 Workshop on Deep Learning for Artificial Intelligence (DLAI), New York, NY, USA, 9–15 July 2016.
13. Yurur, O.; Liu, C.H.; Liu, X.; Moreno, W. Adaptive Sampling and Duty Cycling for Smartphone Accelerometer. In Proceedings of the 2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems, Hangzhou, China, 14–16 October 2013.
14. Bosch, S.; Marin-Perianu, R.; Havinga, P.; Marin-Perianu, M. Energy-Efficient Assessment of Physical Activity Level Using Duty-Cycled Accelerometer Data. *Procedia Comput. Sci.* **2008**, *5*, 328–335. [CrossRef]
15. Gkoktsi, K.; Giaralis, A. A multi-sensor sub-Nyquist power spectrum blind sampling approach for low-power wireless sensors in operational modal analysis applications. *Mech. Syst. Signal Process.* **2019**, *116*, 879–899. [CrossRef]
16. Pang, Z.; Yuan, M.; Wakin, M.B. A random demodulation architecture for sub-sampling acoustic emission signals in structural health monitoring. *J. Sound Vib.* **2018**, *431*, 390–404. [CrossRef]
17. Kumble, L.; Patil, K.K. An Improved Data Compression Framework for Wireless Sensor Networks Using Stacked Convolutional Autoencoder (S-CAE). *SN Comput. Sci.* **2023**, *4*, 419. [CrossRef]
18. ADXL 1002 Datasheet. Available online: https://www.analog.com/media/en/technical-documentation/data-sheets/adxl1001-1002.pdf (accessed on 30 October 2024).

19. CMA-4544PF-W Datasheet. Available online: https://cdn-shop.adafruit.com/datasheets/CMA-4544PF-W.pdf (accessed on 30 October 2024).
20. How Do I Eliminate Ghosting from My Measurements? Available online: https://knowledge.ni.com/KnowledgeArticleDetails?id=kA03q000000YHy6CAG&l=en-US (accessed on 30 October 2024).
21. SigMF Specifications. Available online: https://sigmf.org/index.html (accessed on 30 October 2024).
22. Byington, C.S.; Orsagh, P.R.; Kallappa, P.; Sheldon, J.; DeChristopher, M.; Amin, S.; Hines, J. Recent case studies in bearing fault detection and prognosis. In Proceedings of the 2006 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2006.
23. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *arXiv* **2019**, arXiv:1801.04381.
24. Wi-Fi CERTIFIED HaLow Technology Overview (2021). Available online: https://www.wi-fi.org/file/wi-fi-certified-halow-technology-overview-2021 (accessed on 30 October 2024).
25. 802.11ac-2013—IEEE Standard for Information technology–Telecommunications and information exchange between systemsLocal and metropolitan area networks–Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz. IEEE: New York, NY, USA, 2013. Available online: https://ieeexplore.ieee.org/document/6687187 (accessed on 30 October 2024).
26. 2021 Intermediatory 5G Service Coverage and Quality Evaluation Results. Available online: https://www.msit.go.kr/bbs/view.do?sCode=user&mId=113&mPid=238&bbsSeqNo=94&nttSeqNo=3180664 (accessed on 11 November 2024).
27. Vora, A.; Amipara, K.; Modi, S.; Zaveri, M.A. Data Driven Dynamic Sensor Selection in Internet of Things. In Proceedings of the TENCON 2019—2019 IEEE Region 10 Conference (TENCON), Kochi, India, 17–20 October 2019.
28. MAX4465-69 Data Sheet. Available online: https://www.analog.com/media/en/technical-documentation/data-sheets/MAX4465-MAX4469.pdf (accessed on 11 November 2024).
29. ADS1256 Data Sheet. Available online: https://www.ti.com/lit/ds/symlink/ads1256.pdf?ts=1731303454278&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252Fja-jp%252FADS1256 (accessed on 11 November 2024).
30. STM32F051 Data Sheet. Available online: https://www.st.com/resource/en/datasheet/stm32f051c4.pdf (accessed on 11 November 2024).
31. Tang, S.; Yuan, S.; Zhu, Y. Cyclostationary Analysis towards Fault Diagnosis of Rotating Machinery. *Processes* **2020**, *8*, 1217. [CrossRef]
32. Ahmad, S.; Styp-Rekowski, K.; Nedelkoski, S.; Kao, O. Autoencoder-based Condition Monitoring and Anomaly Detection Method for Rotating Machines. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020.
33. Faber, K.; Corizzo, R.; Sniezynski, B.; Japkowicz, N. Lifelong Continual Learning for Anomaly Detection: New Challenges, Perspectives, and Insights. *IEEE Access* **2024**, *12*, 41364–41380. [CrossRef]

sensors

Article

Mobile Accelerometer Applications in Core Muscle Rehabilitation and Pre-Operative Assessment

Aleš Procházka [1,2,*], Daniel Martynek [1,2], Marie Vitujová [3], Daniela Janáková [3], Hana Charvátová [4] and Oldřich Vyšata [5]

[1] Department of Mathematics, Informatics and Cybernetics, University of Chemistry and Technology in Prague, 166 28 Prague 6, Czech Republic; daniel.martynek@vscht.cz
[2] Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, 160 00 Prague 6, Czech Republic
[3] Department of Sports Medicine, 2nd Faculty of Medicine and FN Motol, Charles University in Prague, 150 00 Prague 5, Czech Republic; marie.vitujova@fnmotol.cz (M.V.); daniela.janakova@fnmotol.cz (D.J.)
[4] Centre for Security, Information and Advanced Technologies (CEBIA-Tech), Faculty of Applied Informatics, Tomas Bata University in Zlín, 760 01 Zlín, Czech Republic; charvatova@utb.cz
[5] Department of Neurology, Faculty of Medicine in Hradec Králové, Charles University in Prague, 500 05 Hradec Králové, Czech Republic; oldrich.vysata@fnhk.cz
* Correspondence: ales.prochazka@vscht.cz or ales.prochazka@cvut.cz

Citation: Procházka, A.; Martynek, D.; Vitujová, M.; Janáková, D.; Charvátová, H.; Vyšata, O. Mobile Accelerometer Applications in Core Muscle Rehabilitation and Pre-Operative Assessment. *Sensors* **2024**, *24*, 7330. https://doi.org/10.3390/s24227330

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 16 October 2024
Revised: 12 November 2024
Accepted: 14 November 2024
Published: 16 November 2024

Abstract: Individual physiotherapy is crucial in treating patients with various pain and health issues, and significantly impacts abdominal surgical outcomes and further medical problems. Recent technological and artificial intelligent advancements have equipped healthcare professionals with innovative tools, such as sensor systems and telemedicine equipment, offering groundbreaking opportunities to monitor and analyze patients' physical activity. This paper investigates the potential applications of mobile accelerometers in evaluating the symmetry of specific rehabilitation exercises using a dataset of 1280 tests on 16 individuals in the age range between 8 and 75 years. A comprehensive computational methodology is introduced, incorporating traditional digital signal processing, feature extraction in both time and transform domains, and advanced classification techniques. The study employs a range of machine learning methods, including support vector machines, Bayesian analysis, and neural networks, to evaluate the balance of various physical activities. The proposed approach achieved a high classification accuracy of 90.6% in distinguishing between left- and right-side motion patterns by employing features from both the time and frequency domains using a two-layer neural network. These findings demonstrate promising applications of precise monitoring of rehabilitation exercises to increase the probability of successful surgical recovery, highlighting the potential to significantly enhance patient care and treatment outcomes.

Keywords: physical activity monitoring; motion symmetry; rehabilitation; abdominal wall repair; computational intelligence; accelerometers; machine learning

1. Introduction

Human activity recognition [1,2] and artificial intelligence (AI) [3] have a wide range of applications in rehabilitation, neurology, and sports. Wearable sensors are widely used in individual physiotherapy, detection of various types of pain, and improvements in physical fitness [4]. Specialized rehabilitation exercises and physical activities [5] play a crucial role in the pre-operative and post-operative stages of surgical treatment [6,7] to optimize the healing and recovery process. This area is increasingly important in line with population ageing, with demands for general surgery expected to rise. However, post-operative complications following abdominal surgery are frequently reported.

Pre-operative assessment is crucial for optimizing surgical outcomes and minimizing post-operative complications, despite the overall success rate sometimes being limited [8].

Traditional pre-operative evaluation methods primarily rely on medical history, physical examination, and imaging studies, which may not always capture the dynamic physiological changes occurring in a patient's daily activities. Various studies show positive effects of rehabilitation in patients undergoing orthopedic surgery [9] and on recovery after abdominal surgery [10]. Deep learning models have been developed to predict rare but severe post-operative complications following specific surgical treatments [11,12].

Different studies focus on prehabilitation exercises and their evaluation [13–16] by computational methods, and their monitoring via telemedicine equipment [17,18] to reduce complication rates and risk factors associated with complex abdominal surgeries. Recent advancements in inertial measurement units (IMUs) for motion capture have introduced novel approaches to pre-operative assessment [7,19,20] and general rehabilitation, with thermal cameras [21] and mobile accelerometers emerging as promising tools in this domain. Mobile accelerometers, commonly found in smartphones and wearable devices [22–25], can continuously monitor a patient's movements, providing real-time data on physical activity, posture, and mobility. This wealth of information offers a unique opportunity to enhance the pre-operative assessment process, enabling a more comprehensive understanding of a patient's functional capacity and aiding surgeons in tailoring their decisions and approach to abdominal wall repair.

The use of mobile accelerometers in pre-operative assessment allows for the collection of objective and quantitative data on a patient's movement patterns and activity levels. By continuously monitoring these metrics, healthcare providers can gain insights into a patient's functional capacity, identify specific movement patterns, and pinpoint high-risk patients. Various rehabilitation programs [26–28] study the effectiveness of specific exercises involving repetitive muscle contractions, core stability, and balance exercises. Further studies explore the use of force sensors to monitor respiratory functions and measure the activation of abdominal wall muscles [29].

Utilizing mobile accelerometers for core muscle rehabilitation and pre-operative assessment involves two key phases: preparing the patients for surgery by assessing their core strength and stability, and aiding in their post-surgery recovery by monitoring and guiding their rehabilitation. By integrating wearable sensors with artificial intelligence tools, clinicians can now assess and monitor patient movements more precisely, allowing for personalized rehabilitation plans.

Mobile accelerometers can play a significant role in both pre-operative and post-operative rehabilitation. By tracking a patient's progress in real-time, rehabilitation protocols can be personalized, monitored remotely, and adjusted based on the patient's response to therapy. This enables more efficient and effective rehabilitation, potentially reducing recovery time, minimizing complications, and improving overall patient satisfaction.

Data processing methods are based on the general methodology of signal processing [30], computational intelligence [31], and time-frequency signal analysis. This approach evaluates features for assessing the balance criterion associated with individual rehabilitation exercises. The classification of symmetry in rehabilitation [32,33] can vary depending on the specific motion parameters [34] being considered and the clinical context. Evaluating separate rehabilitation exercises based on the development of sensor technology has been crucial in realizing the potential for both clinical and remote rehabilitation. While there is no fixed number of symmetry degrees universally used, an objective scoring system can be proposed to evaluate the feasibility of balance assessment technology for adaptation into remote rehabilitation settings. Specific exercises can be proposed for both prehabilitation before different kinds of abdominal surgeries and to treat various motion disorders [35–37], utilizing the important research area of body kinematics [38].

The integration of mobile accelerometers into the rehabilitation and pre-operative assessment process [39] has the potential to enhance interaction between patients, rehabilitation specialists, and surgeons. By leveraging this technology, healthcare providers can gain deeper insights into a patient's daily life and functional capabilities, enabling them to make more informed decisions and provide personalized care. As we explore the role of mobile accelerometers in the context of complex abdominal surgeries [40–43], including open, robotic, and laparoscopic techniques, we will uncover the transformative impact they can have on patient outcomes and the future of surgical practice.

There are many factors that influence ody motion symmetry and asymmetrical movement patterns. They mainly include injuries, muscle strength, age, and neurological disorders that can impact muscle control and coordination, resulting in asymmetry due to impaired movement on one side of the body.

The goal of this paper is to discuss the benefits of using mobile accelerometers in the evaluation of rehabilitation exercises to reduce the probability of complications after surgeries. Associated topics include the abilities of this methodology to capture objective and quantitative data, track changes in physical activity levels, and detect movement patterns. Another goal is to address the challenges and limitations associated with this technology, such as data privacy concerns, device compatibility, and the need for standardized algorithms to interpret accelerometer data. The proposed data analysis procedures contribute to this research area by (i) demonstrating the use of smartphones, communication links, and remote data stores to record accelerometric data during rehabilitation exercises, (ii) proposing a fast symmetry level evaluation using a suggested global criterion function, and (iii) designing a general web-page that allows data import, remote signal processing in both time and frequency domains, and evaluation of the coefficient of symmetry. The novelty of the paper lies in the use of communication links for data acquisition with remote storage and the proposal of a symmetry coefficient evaluation.

2. Methods

This paper describes the use of wearable accelerometers to analyze the symmetry of different rehabilitation exercises using wearable sensors embedded in a mobile phone [44,45] placed on the body. Figure 1 illustrates the framework for analyzing rehabilitation exercises and data processing that include

(a) Activation of sensors in a smartphone and specification of their parameters in the mobile Matlab environment.
(b) Data acquisition from the right and left part of the body with the selected sampling frequency.
(c) Export of signals through communication links into the remote drive.
(d) Evaluation of accelerometric signals, estimation of the symmetry coefficient of left/right parts of the body, and classification of motion features.

The selected rehabilitation exercises were acquired and processed in the Matlab 2024b (MathWorks, Natick, MA, USA) computational environment. Data were recorded by mobile Matlab connected to the Matlab cloud with saving data on the Matlab Drive.

The dataset includes records acquired by a smartphone equipped with a three-axis accelerometer. All procedures involving human participants were conducted in accordance with the ethical standards of the institutional research committee and the 1964 Helsinki Declaration and its later amendments. The study received ethical approval from the Ethics Committee (UCT EK/7/2022), and the anonymity of the obtained data was strictly maintained.

The analysis is based on eight exercises performed during 1280 tests on different individuals. Detailed descriptions of observations can be found on IEEE DataPort (Rehabilitation Exercises and Computational Intelligence, 10.21227/xp41-7325) [46] for further investigation. This repository includes the accelerometric data acquired during all experiments, an informative video presentation of the rehabilitation exercises, the Matlab graphical user interface, and a graphical video abstract of the paper.

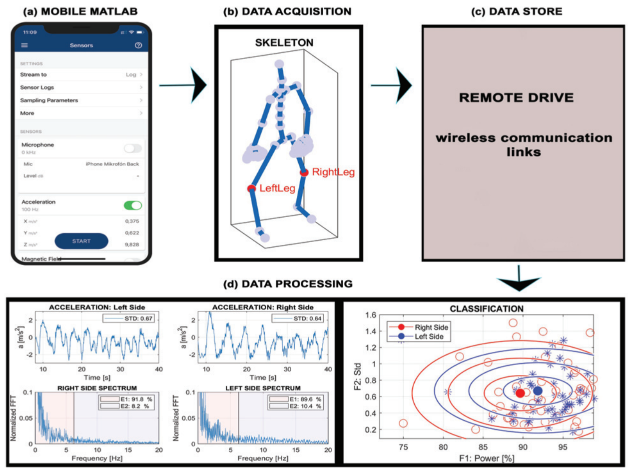


Figure 1. Principle of data processing during rehabilitation exercises presenting (**a**) mobile Matlab initialization, (**b**) data acquisition using accelerometric sensors inside the smartphone, (**c**) export of recorded signals to the remote drive, and (**d**) processing of data on the remote drive in time and frequency domains to extract motion features and evaluate the coefficient of symmetry.

2.1. Data Acquisition

Figure 2 and Table 1 present a brief specification of the selected rehabilitation exercises. The smartphone was affixed to the left or right leg or arm with the display facing forward [47] and was used as a sensor for accelerometric data acquisition via the mobile Matlab application, with a sampling frequency of 100 Hz. Signals from the left and right sides of the body were acquired and processed in the Matlab environment. Each exercise was repeated ten times and performed during 16 tests involving different individuals.

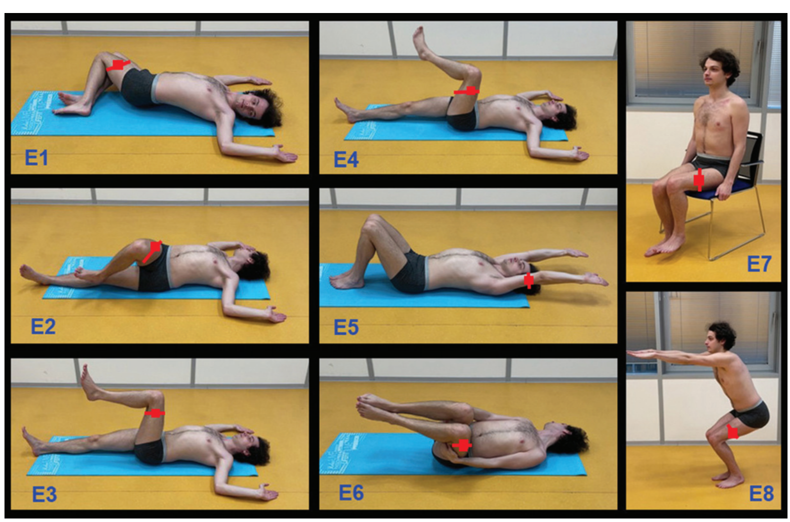


Figure 2. Selected rehabilitation exercises used for accelerometric data acquisition recorded by wearable sensors (red squares) located on the left and right sides of the body used for data acquisition and processing in the computational and visualization environment of the mobile Matlab system.

Table 1. Description of selected exercises used for prehabilitation before the surgery treatment.

Exercise	Name	Description
E1	basic spinal motion	both legs bent
E2	spinal motion	one leg bent
E3	lifting of one leg	other leg on the floor
E4	foot circles	circles in the hip joint
E5	arm flection	arms motion
E6	body cross-motion	body sculpture rotation
E7	leg lifting	one-leg lift
E8	squat	high squat

The study included 16 participants, comprising 9 males and 7 females, with ages ranging from 8 to 75 years and BMI value 23.4 ± 2.9 kg/m^2. Detailed participant information is presented in Table 2.

Table 2. Description of participants in the rehabilitation exercises, including age, gender, height, and BMI of each individual.

Individual	Age [year]	Gender m/f	Height [cm]	BMI [kg/m^2]
1-AP	75	m	187	27.7
2-HCH	45	f	152	21.6
3-AM	21	f	173	18.0
4-DM	21	m	184	21.6
5-DDM	47	m	178	26.5
6-DH	24	m	185	22.8
7-JH	21	m	176	22.3
8-JM	69	m	185	27.5
9-LN	22	m	182	19.0
10-VM	47	f	163	25.6
11-MS	34	m	192	27.1
12-AB	47	m	176	22.6
13-TT	22	f	175	24.5
14-KA	8	f	135	21.6
15-T2	22	f	175	24.5
16-H2	46	f	152	21.6
MEAN	35.7		173.1	23.4
STD	18.9		15.3	2.9

2.2. Signal Processing

In the field of rehabilitation, accelerometric data processing is essential for monitoring and analyzing movement patterns. Computational intelligence tools play a significant role in this domain by providing advanced methods for data analysis, interpretation, and decision-making, which aid in developing personalized rehabilitation programs.

Fundamental signal processing methods include digital filtering techniques to remove noise and extract relevant signal components. The Fourier transform converts accelerometric signals from the time domain to the frequency domain, facilitating the analysis of periodic components. Alternatively, the Wavelet transform offers multi-resolution analysis of accelerometric signals, enabling the detection of both transient and continuous features. Machine learning algorithms, such as support vector machines, decision trees, and neural networks, are then employed to classify movement patterns and predict rehabilitation outcomes.

Three-dimensional accelerometers are widely used in rehabilitation to monitor and assess body movement comprehensively, using acceleration data across three orthogonal axes (x, y, and z). The resulting data, observed with the sampling frequency f_s, form column vectors $\mathbf{d}_p$ for position p, representing the left ($p = L$) and right ($p = R$) parts of the body. Each sequence can then be divided into M subsequences, each N values long, defining a vector $[\mathbf{d}_L^{(1)}, \cdots, \mathbf{d}_L^{(M)}, \mathbf{d}_R^{(1)}, \cdots, \mathbf{d}_R^{(M)}]$ and forming a time-domain signal matrix:

$$\mathbf{D}_{N\times Q} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \cdots & \mathbf{d}_Q \end{bmatrix} \tag{1}$$

with $Q = 2 \times M$ columns. The elements of each column vector $\mathbf{d}_q = \{d_q(n)\}_{n=(k-1)N+1}^{kN}$ for $q = 1, 2, \cdots, Q$ specify the observed accelerometric data.

By applying the discrete Fourier transform (dft) to each column of the matrix $\mathbf{D}_{N\times Q}$, we can construct the associated matrix $\mathbf{G}_{N\times Q}$ as follows:

$$\mathbf{G}_{N\times Q} = dft(\mathbf{D}_{N\times Q}) \tag{2}$$

This matrix contains the frequency components of the separate signal subwindows in its column vectors, with frequency components ranging from 0 to f_s Hz. The values in each column q of the matrix $\mathbf{G}_{N\times Q}$ are evaluated by the following relation:

$$g_q(k) = \sum_{n=1}^{N} (d_q(n) - \bar{d}_q)\; e^{-j\,(k-1)\,(n-1)\,2\,\pi/N} \tag{3}$$

for $k = 1, 2, \cdots, N$, and the mean value $\bar{d}_q$ of each column q of the matrix $\mathbf{D}_{N\times Q}$ for a selected individual and the left and right sides of the body. Spectral components are evaluated with the frequency resolution $\frac{1}{N} f_s$ Hz and its values $f(k) = \frac{k}{N} f_s$.

With the frequency components of each accelerometric signal subwindow, it is possible to evaluate the relative power $E_q^{(i)}$ for each subwindow q of the observed sequence in the frequency band $B^{(i)} = \langle f_{c_1}^{(i)}, f_{c_2}^{(i)} \rangle$ using spectral components evaluated by the discrete Fourier transform according to Equation (3) by the relation:

$$E_q^{(i)} = \frac{\sum_{n\in\Phi^{(i)}} |g_q(n)|^2}{\sum_{n=1}^{N/2} |g_q(n)|^2} \tag{4}$$

where $\Phi^{(i)}$ is the set of indices for the frequency components $f(k)^{(i)} \in \langle f_{c_1}^{(i)}, f_{c_2}^{(i)} \rangle$. This process can be applied for the whole matrix $\mathbf{G}_{N\times Q}$ to find Q features of separate subwindows and selected frequency bands.

When analyzing a selected rehabilitation exercise, each subwindow can be described (i) in the time domain by its mean and standard deviation, and (ii) in the frequency domain by the power in the selected frequency bands. This forms a matrix with time- and frequency-domain features for the left and right side of the body forming a vector $[\mathbf{p}_L^{(1)}, \cdots, \mathbf{p}_L^{(M)}, \mathbf{p}_R^{(1)}, \cdots, \mathbf{p}_R^{(M)}]$.

Matrix $\mathbf{D}_{N\times Q}$ is used for the evaluation of means and standard deviations in the time domain, as well as to evaluate power components to form a pattern matrix:

$$\mathbf{P}_{R\times Q} = [\mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_Q}] \tag{5}$$

that includes R features evaluated for each subwindow associated with the rehabilitation experiment.

The associated target vector specifying the positions of sensors includes the values $[L, \cdots, L, R, \cdots, R]$ that can be substituted by target probabilities of each class:

$$\mathbf{T}_{S\times Q} = [\mathbf{t_1}, \mathbf{t_2}, \cdots, \mathbf{t_Q}] = \begin{bmatrix} 0 & \cdots & 0 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 0 & \cdots & 0 \end{bmatrix} \tag{6}$$

for classification into two classes (S = 2) associated with the positions of the sensors on the body during each rehabilitation exercise.

Motion symmetry is a valuable concept in analyzing rehabilitation exercises because it allows clinicians and researchers to evaluate whether movements on both sides of the body are aligned, balanced, and coordinated, which is essential for assessing the recovery progress. During rehabilitation, symmetry in motion between the left and right sides of the body is often a goal, particularly before and after surgeries using specific motion capture systems or wearable sensors.

Complete records of the accelerometric signal for the left and right sides of the body were divided into M segments and for each of them, the evaluation was performed to define the left-side feature $F_q^{(L)}(r)$ and the right-side feature $F_q^{(R)}(r)$ based on property r for segment $q = 1, 2, \cdots, M$. The symmetry index, based on a commonly used one, can be calculated by the following relation:

$$c_q(r) = \frac{1}{2} \frac{F_q^{(L)}(r) - F_q^{(R)}(r)}{F_q^{(L)}(r) + F_q^{(R)}(r)} 100 \tag{7}$$

The average of $c_q(r)$ over all segments $q \in \langle 1, Q \rangle$ results in the standard symmetry coefficient related to the selected feature r.

The alternative criterion for one experiment and a selected feature set can be evaluated using the proposed relation:

$$C = \sqrt{\frac{1}{length(\Psi)} \sum_{r \in \Psi} \frac{1}{Q} \sum_{q=1}^{Q} c_q(r)} \tag{8}$$

where Ψ is the selected set of features using the global symmetry criterion evaluation.

Classifying rehabilitation exercises using accelerometers involves several steps, including data collection, preprocessing, feature extraction, and classification. The classification of Q signal segments by a specific machine learning method typically requires the determination of the pattern and target matrices. In this case, the pattern matrix $\mathbf{P}_{R,Q}$ defined by Equation (5), and target matrix $\mathbf{T}_{S,Q}$ specified by Equation (6), respectively, were used. The number of features was reduced to $R = 2$ for better visualization.

Commonly used algorithms for signal segment classification include support vector machines (SVM), which are effective in high-dimensional spaces, Bayesian methods, and the simple and commonly used k-nearest neighbor methods. Alternatively, neural network methods, including deep learning approaches, are suitable for handling large and complex systems. In the simplest case of a two-layer neural network with S_1 and S_2 elements in the first and second layers, respectively, the outputs $\mathbf{A}_{S_1,Q}^{(1)}$ and $\mathbf{A}_{S_2,Q}^{(2)}$ of the individual layers are evaluated by the following relations:

$$\begin{aligned} \mathbf{A}_{S_1,Q}^{(1)} &= f_1(\mathbf{W}_{S_1,R}^{(1)}\ \mathbf{P}_{R,Q}, \mathbf{b}_{S_1,1}^{(1)}) \\ \mathbf{A}_{S_2,Q}^{(2)} &= f_2(\mathbf{W}_{S_2,S_1}^{(2)}\ \mathbf{A}_{S_1,Q}^{(1)}, \mathbf{b}_{S_2,1}^{(2)}) \end{aligned} \tag{9}$$

with the network coefficients forming matrices $\mathbf{W}_{S_1,R}^{(1)}$ and $\mathbf{W}_{S_2,S_1}^{(2)}$, and vectors $\mathbf{b}_{S_1,1}^{(1)}$ and $\mathbf{b}_{S_2,1}^{(2)}$. The proposed model uses the sigmoidal transfer function f_1 in the first layer and the probabilistic softmax transfer function f_2 in the second layer.

To determine the predictive model's ability to perform classification during practical implementation, the k-fold cross-validation method is often used. In this paper, the leave-one-out method, with the same number of folds as the number of data points, is employed.

When implementing and evaluating models, it is crucial to consider the context and the specific costs associated with false positives and false negatives. Sensitivity and specificity provide a clear picture of the model's performance in identifying both positive and negative cases, aiding in making decisions about model deployment and potential improvements. For classifying rehabilitation exercises using accelerometers into two classes, common performance metrics can be used:

- Sensitivity (True positive rate, recall) defined as the proportion of actual positives that are correctly identified by relation:

$$TPR = \frac{TP}{TP + FN} \quad (10)$$

- Specificity (True negative rate) defined as the proportion of actual negatives that are correctly identified by relation:

$$TNR = \frac{TN}{TN + FP} \quad (11)$$

- Accuracy defined as a probability of global correct classification:

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \quad (12)$$

where TP, TN, FP and FN stand for the number of true positive, true negative, false positive, and false negative classifications [48].

3. Results

The proposed graphical user interface [49,50] is presented in Figure 3. It enables the visualization of rehabilitation exercises through videos accessible from the initial web-page. The motion accelerometric data acquisition and processing using a specific web-page includes the following steps:

- Animating motion exercises for training and data acquisition by a mobile phone.
- Selecting accelerometric signals recorded by the smartphone of a chosen individual and stored in the specified datastore.
- Trimming inaccurate data at the beginning and end of each record.
- Evaluating spectral components recorded on the right and left sides of the body using the discrete Fourier transform, with results displayed in Figure 3b.
- Estimating the percentage power of signals in selected frequency ranges and specified subwindows.
- Visualizing motion features associated with the left and right sides of the body.
- Evaluating the proposed symmetry criterion coefficient for the selected rehabilitation exercise.

Table 3 lists the individuals, types of rehabilitation exercises, and the symmetry criterion values evaluated by Equation (8) using features in the frequency domain for each exercise. The last column includes the average symmetry index values for each individual. The last two rows present the overall average symmetry coefficients across all individuals and their standard deviations.

Figure 4 presents the symmetry criteria for eight rehabilitation exercises, evaluated using both time domain and spectral domain features. It shows the mean values of 16 tests, each with 10 repetitions of each rehabilitation exercise for a selected individual. The highest asymmetry, exceeding the mean value, was observed for exercises E2, E3, E4, and E6 by both methods. The best symmetry criterion coefficients were observed for exercises E1 and E7.

More detailed results are presented in Table 3 for the set of individuals under study. A comparison of symmetry criteria for 16 tests involving different individuals and 8 rehabilitation exercises, evaluated using time domain and spectral domain features, is shown in Figure 5. The best symmetry was observed for individual 15, with a mean symmetry coefficient of 1.1 across all rehabilitation exercises.

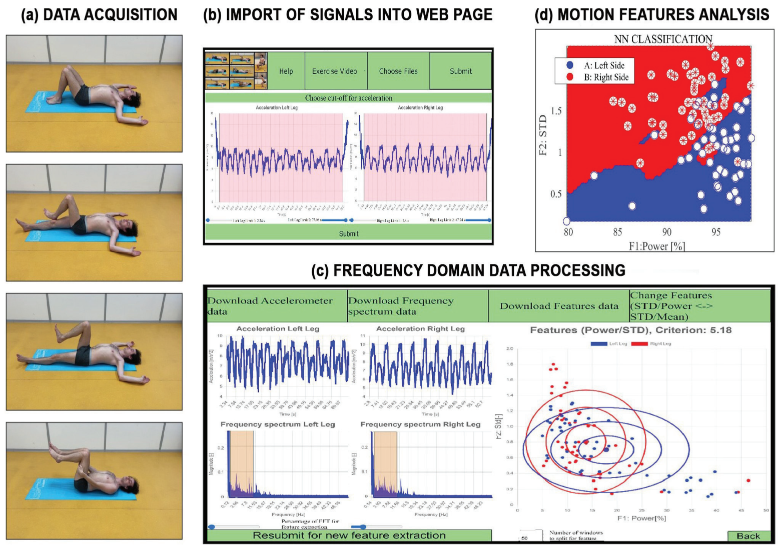


Figure 3. Principle of data processing during rehabilitation exercises presenting (**a**) animation of motion exercises to train individuals and data acquisition using a smartphone, (**b**) data import into the proposed web-page, (**c**) frequency domain remote signal processing including symmetry coefficient estimation, and (**d**) extraction and analysis of motion features.

Table 3. Results of symmetry values of the set of 16 tests of different individuals (*Ind*) for separate exercises evaluated by the mixed-domain features, and their mean values associated with each participant of the study.

Ind.	Exercise								Mean
	E1	E2	E3	E4	E5	E6	E7	E8	
1	1.5	1.7	2.1	2.4	2.3	7.7	1.0	5.7	3.0
2	1.5	3.7	4.5	2.1	1.1	0.6	2.9	3.7	2.5
3	2.0	0.8	1.9	3.2	3.5	7.3	1.6	1.6	2.7
4	3.0	0.8	1.8	1.4	0.3	3.5	3.5	3.3	2.2
5	3.1	6.2	6.7	3.8	3.8	2.9	1.6	0.6	3.6
6	4.8	3.6	5.1	5.8	3.6	7.0	1.3	3.0	4.3
7	0.5	0.7	1.5	3.9	2.5	1.3	0.8	1.9	1.6
8	3.9	3.1	3.7	0.8	3.2	3.8	4.2	7.6	3.8
9	2.9	4.8	5.9	4.7	5.9	2.5	1.5	0.9	3.6
10	2.5	5.7	0.8	5.6	0.1	4.0	2.3	0.5	2.7
11	0.1	3.9	4.7	2.3	2.9	1.2	1.6	4.4	2.6
12	1.3	2.5	1.8	1.1	2.0	0.8	4.5	3.0	2.1
13	0.7	3.5	0.8	4.0	1.3	4.7	1.8	1.6	2.3
14	4.3	0.6	1.4	7.5	3.6	4.6	7.3	3.4	4.1
15	1.0	1.0	2.0	1.0	1.3	0.4	0.8	1.3	1.1
16	1.4	1.7	4.0	3.6	0.6	1.7	1.3	5.5	2.5
Mean	2.1	2.8	3.0	3.3	2.4	3.4	2.3	3.0	
Std	1.4	1.8	1.9	1.9	1.6	2.4	1.8	2.0	

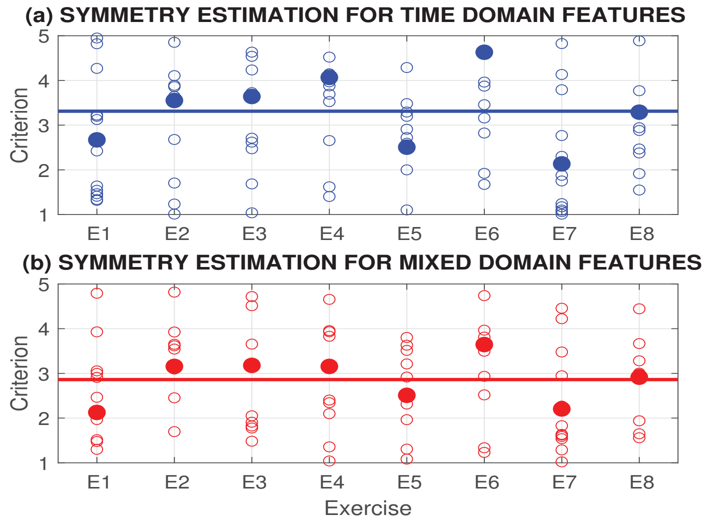


Figure 4. Symmetry criteria for 8 rehabilitation exercises evaluated by (**a**) time domain and (**b**) mixed-domain features presenting mean values by 16 tests of different individuals with 10 repetitions of each rehabilitation exercise.

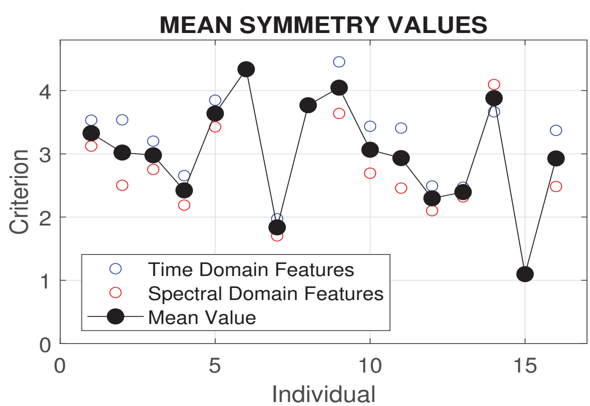


Figure 5. Comparison of symmetry criteria for 16 tests involving different individuals and eight rehabilitation exercises, evaluated using time domain and spectral domain features.

Features evaluated from the sensors on the left and right sides of the body for selected exercises, as well as mixed-domain features, are presented in Figure 6. This comparison shows results for selected exercises that demonstrate prevailing asymmetric and symmetric motions, with centers of the right and left side positions and multiples of standard deviations for $c = 0.2, 0.5, 1$.

Figure 7 presents the classification of the symmetry features for cross-motion (exercise 6) using mixed features and three different methods: support vector machine, the Bayesian method [51,52], and a two-layer neural network (NN) with 10 neurons in the first layer and sigmoidal/softmax transfer functions in the first and second layers, respectively.

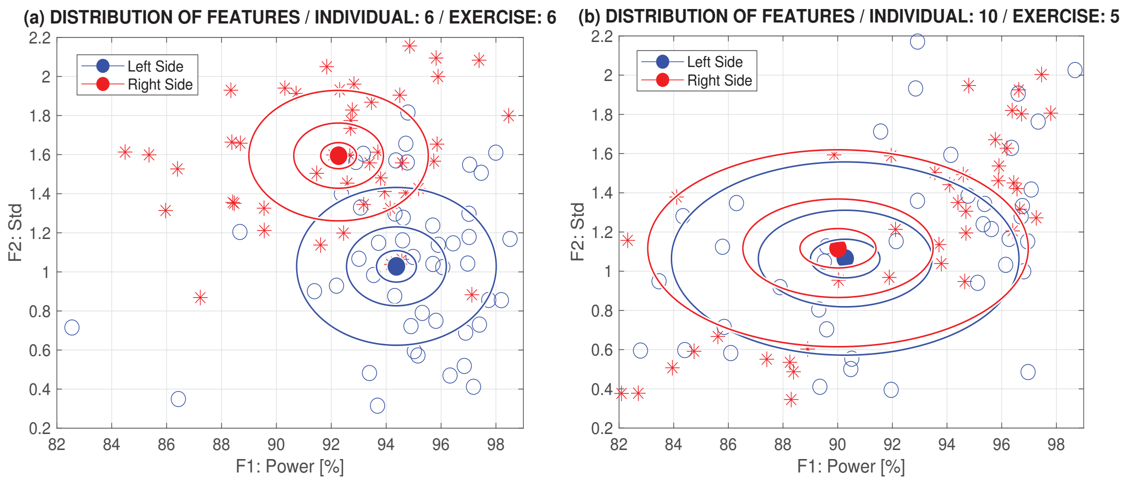


Figure 6. Comparison of distribution of the time and spectral domain features for selected exercises of (**a**) prevailing asymmetric motion (individual 6, exercise 6) and (**b**) prevailing symmetric motion (individual 10, exercise 5) with centers of the right and left side positions and c multiples of standard deviations for $c = 0.2, 0.5, 1$.

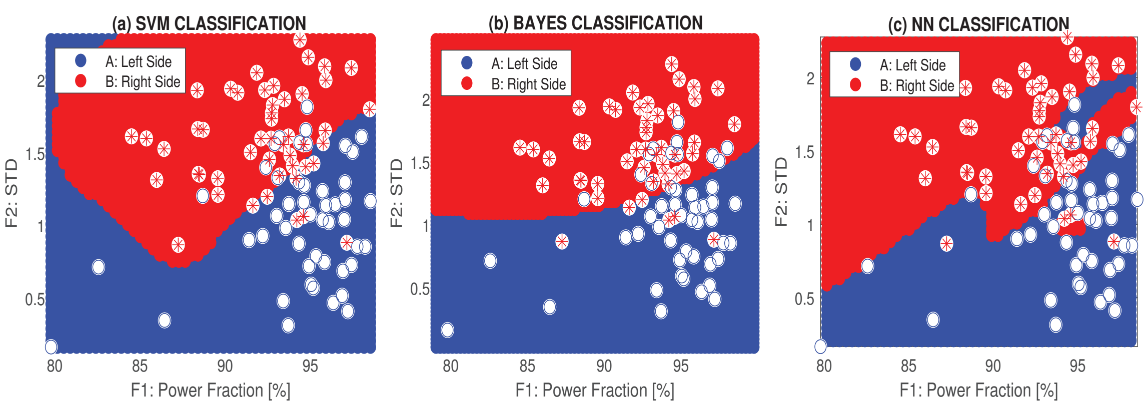


Figure 7. Classification of symmetry features of the body cross-motion by mixed features using (**a**) support vector machine, (**b**) the Bayes method, and (**c**) the two-layer neural network for a selected individual 6-DH.

A summary of the accuracy and cross-validation errors for all individuals, a selected rehabilitation exercise, and different classification methods is presented in Table 4. The highest classification accuracy of 90.6% for individual 6-DH corresponds with their worst coefficient of symmetry in Table 3. The cross-validation errors were calculated using the leave-one-out method.

Table 4. Symmetry classification of rehabilitation exercise patterns performed by the support vector machine, Bayesian, and two-layer neural network methods using two features specified as the power in the selected frequency band and the associated standard deviation of accelerometric data, presenting the accuracy (AC) and the cross-validation error (CV) for exercise 6 and all individuals (*Ind*).

Ind.	SVM Method		Bayes Method		NN Method	
	AC [%]	CV	AC [%]	CV	AC [%]	CV
1	69.6	0.39	59.8	0.37	76.1	0.34
2	72.0	0.42	54.8	0.53	76.3	0.16
3	84.9	0.25	79.6	0.28	84.9	0.16
4	63.4	0.45	54.8	0.56	66.7	0.44
5	76.8	0.37	73.7	0.23	82.1	0.18
6	86.5	0.21	81.3	0.25	90.6	0.11
7	72.6	0.38	71.6	0.35	72.6	0.21
8	63.4	0.45	59.1	0.48	74.2	0.25
9	66.3	0.40	64.1	0.47	78.3	0.30
10	73.7	0.39	63.2	0.40	75.8	0.20
11	68.1	0.44	52.7	0.48	69.2	0.22
12	60.9	0.48	52.2	0.47	72.8	0.23
13	68.5	0.46	64.1	0.34	70.7	0.35
14	71.4	0.31	72.5	0.33	81.3	0.21
15	69.1	0.43	50.0	0.39	66.0	0.39
16	72.3	0.33	67.0	0.38	76.6	0.32

4. Discussion

This paper focuses on evaluating rehabilitation exercises designed to strengthen the abdominal wall and reduce complications during potential surgeries. Accelerometric data from a variety of exercises were used for symmetry analysis of different motion patterns across 16 individuals. The features obtained were evaluated in both the time and spectral domains to classify rehabilitation segments and various exercises. This approach is significant for improving body fitness levels, reducing potential chest pains, and serving as pre-operative muscle training before open or robotic surgery [53].

The most important features for evaluating the rehabilitation exercises were based on signals recorded by an accelerometer inside a smartphone positioned on a specific part of the body. The worst mean coefficient of symmetry was found for exercise E6, with a mean value of 3.4 and the highest standard deviation of 2.4, indicating the difficulty of this rehabilitation motion for all participants. Exercises E1, E2, E5, and E7 were among the easier ones, with their coefficients of symmetry below 3 and standard deviations below 2.

The classification accuracy reached 90.6% for the two-layer neural network, with a cross-validation error of 0.11 (using the leave-one-out method) for individual 6-DH, who had the highest asymmetry and a coefficient of symmetry of 7, as shown in Figure 7. Rehabilitation specialists must also consider individuals' ages, as some exercises are complicated for elderly people, and their physical condition can affect the performance of rehabilitation exercises.

Integrating sensor technology into pre-operative and post-operative care could help develop a more sophisticated data-driven approach to surgical planning, monitoring, and follow-up, improving outcomes and advancing the standards of surgical care. Sensors can continuously track key indicators, such as range of motion and muscle strength. Early signs of complications can be detected through subtle changes in these indicators.

Limiting factors of the use of accelerometers for the evaluation of rehabilitation symmetry include sensitivity to the placement of sensors, sensitivity to external factors (like uneven surfaces), congenital asymmetry of movement, age of patients, and data interpretation. The influence of the last item can be reduced by correctly selected signal processing methodology and AI application.

Evaluation of the rehabilitation exercises can be conducted through the proposed web-page to inform individuals about their progress and to motivate them to perform rehabilitation exercises more precisely. Future studies should focus on more complex computational methods and the use of multichannel sensor systems for time-synchronized monitoring of motion patterns, allowing for a more detailed analysis of rehabilitation exercises. These applications will include the use of more sophisticated sensors, advanced computational methods, and deep learning strategies to monitor rehabilitation patterns, potentially utilizing augmented reality and telerehabilitation.

5. Conclusions

The integration of mobile accelerometers and computational analysis in rehabilitation exercises has the potential to significantly enhance healthcare and fitness levels. The comprehensive data from appropriate sensors can lead to more personalized and data-driven decision-making, improving patient outcomes. To realize the full potential of mobile accelerometers in rehabilitation care, further research and collaboration between healthcare professionals and technology developers are critical.

There is a high risk of developing post-operative complications after abdominal surgery for patients with lower pre-operative physical activity. Hence, pre-operative specific rehabilitation and physical activity measurement may be useful in decreasing post-surgery complications.

The future of computational intelligence in the analysis of rehabilitation exercises is promising, with the potential to significantly enhance the precision, personalization, and effectiveness of rehabilitation programs. An emerging trend involves using wearable sensors and digital technology to monitor motion activities. By integrating advanced technologies such as machine learning, computer vision, and robotics, rehabilitation can become more adaptive, patient-centered, and efficient, ultimately leading to better outcomes and an improved quality of life for patients undergoing surgery.

In rehabilitation settings, the use of accelerometers is becoming more prevalent due to their affordability, ease of use, and integration with other technologies. This makes them a key tool in improving the quality and effectiveness of rehabilitation therapy.

Author Contributions: A.P. was responsible for the mathematical methods of data processing, D.M. recorded all data, M.V. and D.J. were responsible for proposal of the rehabilitation exercises, H.C. contributed to data analysis, and O.V. contributed to data acquisition and evaluation of results. All authors have read and agreed to the published version of the manuscript.

Funding: This investigation was reinforced by the European Union under the project ROBOPROX—Robotics and Advanced Industrial Production (reg.no. CZ.02.01.01/00/22_008/0004590) in the area of machine learning. The research related to data acquisition and their computational processing was supported by Operational Programme Johannes Amos Comenius financed by European Structural and Investment Funds and the Czech Ministry of Education, Youth and Sports (Project No. SENDISO—CZ.02.01.01/00/22_008/0004596).

Institutional Review Board Statement: All procedures involving human participants were conducted in accordance with the ethical standards of the institutional research committee and the 1964 Helsinki Declaration and its later amendments. The study received ethical approval from the Ethics Committee (UCT EK/7/2022), and the anonymity of the obtained data was strictly maintained.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Accelerometric data acquired during all experiments with detailed descriptions of observations can be found on IEEE DataPort (Rehabilitation Exercises and Computational Intelligence, 10.21227/xp41-7325) for further investigation (https://doi.org/10.21227/xp41-7325, accessed on 11 November 2024).

Acknowledgments: Thanks are due to all participants of the study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gomaa, W.; Khamis, M. A perspective on human activity recognition from inertial motion data. *Neural Comput. Appl.* **2023**, *35*, 20463–20568. [CrossRef]
2. Xu, Z.; Wu, Z.; Wang, L.; Ma, Z.; Deng, J.; Sha, H.; Wang, H. Research on Monitoring Assistive Devices for Rehabilitation of Movement Disorders through Multi-Sensor Analysis Combined with Deep Learning. *Sensors* **2024**, *24*, 4273. [CrossRef] [PubMed]
3. Wei, S.; Wu, Z. The Application of Wearable Sensors and Machine Learning Algorithms in Rehabilitation Training: A Systematic Review. *Sensors* **2023**, *23*, 7667. [CrossRef] [PubMed]
4. Carnevale, A.; Longo, U.; Schena, E.; Massaroni, C.; Lo Presti, C.; Berton, A.; Candela, V.; Denaro, V. Wearable systems for shoulder kinematics assessment: A systematic review. *BMC Musculoskelet. Disord.* **2019**, *20*, 546. [CrossRef]
5. Grimes, L.; Outtrim, J.; Griffin, S.; Ercole, A. Accelerometery as a measure of modifiable physical activity in high- risk elderly preoperative patients: A prospective observational pilot study. *BMJ Open* **2019**, *9*, e032346. [CrossRef]
6. Regterschot, G.; Ribbers, G.; Bussmann, J. Wearable Movement Sensors for Rehabilitation: From Technology to Clinical Practice. *Sensors* **2021**, *23*, 4744. [CrossRef]
7. Syversen, A.; Dosis, A.; Jayne, D.; Zhang, Z. Wearable Sensors as a Preoperative Assessment Tool: A Review. *Sensors* **2024**, *24*, 482. [CrossRef]
8. McIsaac, D.; Gill, M.; Boland, L.; Hutton, B.; Branje, K.; Shaw, J.; Grudzinski, A.; Barone, N.; Gillis, C. Prehabilitation in adult patients undergoing surgery: An umbrella review of systematic reviews. *Br. J. Anaesth.* **2022**, *128*, 244–257. [CrossRef]
9. Master, H.; Bley, J.; Coronado, R.; Robinette, P.; White, D.; Pennings, J.; Archer, K. Effects of physical activity interventions using wearables to improve objectively-measured and patient-reported outcomes in adults following orthopaedic surgical procedures: A systematic review. *PLoS ONE* **2022**, *17*, e0263562. [CrossRef]
10. Adams, S.; Bedwani, N.; Massey, L.; Bhargava, A.; Byrne, C.; Jensen, K.; Smart, N.; Walsh, C. Physical activity recommendations pre and post abdominal wall reconstruction: A scoping review of the evidence. *Hernia* **2022**, *26*, 701–714. [CrossRef]
11. Ayuso, S.; Elhage, S.; Zhang, Y.; Aladegbami, B.; Gersin, K.; Fischer, J.; Augenstein, V.; Colavita, P.; Heniford, B. Predicting rare outcomes in abdominal wall reconstruction using image-based deep learning model. *Surgery* **2023**, *173*, 748–755. [CrossRef] [PubMed]
12. Timmer, A.; Claessen, J.; Boermeester, M. Risk Factor-Driven Prehabilitation Prior to Abdominal Wall Reconstruction to Improve Postoperative Outcome. A Narrative Review. *J. Abdom. Wall Surg.* **2022**, *1*, 10722. [CrossRef] [PubMed]
13. Kamarajah, S.; Bundred, J.; Weblin, J.; Tan, B. Critical appraisal on the impact of preoperative rehabilitation and outcomes after major abdominal and cardiothoracic surgery: A systematic review and meta-analysis. *Surgery* **2020**, *167*, 540–549. [CrossRef]
14. Hughes, M.; Hackney, R.; Lamb, P.; Wigmore, S.; Deans, D.; Skipworth, R. Prehabilitation Before Major Abdominal Surgery: A Systematic Review and Meta-analysis. *World J. Surg.* **2019**, *43*, 1661–1668. [CrossRef]
15. Liao, Y.; Vakanski, A.; Xian, M.; Paul, D.; Baker, R. A review of computational approaches for evaluation of rehabilitation exercises. *Comput. Biol. Med.* **2020**, *119*, 721. [CrossRef]
16. Jeske, P.; Wojtera, B.; Banasiewicz, T. Prehabilitation-Current Role in Surgery. *Pol. J. Surg.* **2022**, *94*, 65–72. [CrossRef]
17. Pan, H.; Wang, H.; Li, D.; Zhu, K.; Gao, Y.; Yin, R.; Shull, P. Automated, IMU-based spine angle estimation and IMU location identification for telerehabilitation. *Neural Comput. Appl.* **2024**, *21*, 96. [CrossRef]
18. Lee, A.; Deutsch, J.; Holdsworth, L.; Kaplan, S.; Kosakowski, H.; Latz, R.; McNeary, L.; O'Neil, J.; Ronzio, O.; Sanders, K.; et al. Telerehabilitation in Physical Therapist Practice: A Clinical Practice Guideline From the American Physical Therapy Association. *Phys. Ther.* **2024**, *104*, pzae045. [CrossRef]
19. Abouelnaga, W.; Aboelnour, N. Effectiveness of Active Rehabilitation Program on Sports Hernia: Randomized Control Trial. *Ann. Rehabil. Med.* **2019**, *43*, 305–313. [CrossRef]
20. Gillis, C.; Ljungqvist, O.; Carli, F. Prehabilitation, enhanced recovery after surgery, or both? A narrative review. *Br. J. Anaesth.* **2022**, *128*, 434–448. [CrossRef]
21. Vutan, A.; Lovasz, E.; Gruescu, C.; Sticlaru, C.; Sirbu, E.; Jurjiu, N.; Borozan, I.; Vutan, C. Evaluation of Symmetrical Exercises in Scoliosis by Using Thermal Scanning. *Appl. Sci.* **2022**, *12*, 721. [CrossRef]
22. Whelan, D.; O'Reilly, M.; Ward, T.; Delahunt, E.; Caulfield, B. Technology in Rehabilitation: Evaluating the Single Leg Squat Exercise with Wearable Inertial Measurement Units. *Methods Inf. Med.* **2017**, *56*, 88–94. [PubMed]
23. Basil, G.; Sprau, A.; Eliahu, K.; Borowsky, P.; Wang, M.; Jang, W. Using Smartphone-Based Accelerometer Data to Objectively Assess Outcomes in Spine Surgery. *Neurosurgery* **2021**, *88*, 763–772. [CrossRef] [PubMed]
24. Wang, X.; Yu, H.; Kold, S.; Rahbek, O.; Bai, S. Wearable sensors for activity monitoring and motion control: A review. *Biomim. Intell. Robot.* **2023**, *3*, 100089. [CrossRef]
25. Huang, X.; Xue, Y.; Ren, S.; Wang, F. Sensor-Based Wearable Systems for Monitoring Human Motion and Posture: A Review. *Sensors* **2023**, *23*, 9047. [CrossRef]
26. Prat-Luri, A.; Moreno-Navarro, P.; Carpenac, C.; Manca, A.; Deriu, F.; Barbado, D.; Vera-Garcia, F. Smartphone accelerometry for quantifying core stability and developing exercise training progressions in people with multiple sclerosis. *Mult. Scler. Relat. Disord.* **2023**, *72*, 104618. [CrossRef]

27. Renshaw, S.; Peterson, R.; Lewis, R.; Olson, M.; Henderson, W.; Kreuz, B.; Poulose, B.; Higgins, R. Acceptability and barriers to adopting physical therapy and rehabilitation as standard of care in hernia disease: A prospective national survey of providers and preliminary data. *Hernia* **2022**, *26*, 865–871. [CrossRef]
28. Perez, J.; Schmidt, M.; Narvaez, A.; Welsh, L.; Diaz, R.; Castro, M.; Ansari, K.; Cason, R.; Bilezikian, J.; Hope, W.; et al. Evolving concepts in ventral hernia repair and physical therapy: Prehabilitation, rehabilitation, and analogies to tendon reconstruction. *Hernia* **2021**, *25*, 1–13. [CrossRef]
29. Novak, J.; Busch, A.; Kolar, P.; Kobesova, A. Postural and respiratory function of the abdominal muscles: A pilot study to measure abdominal wall activity using belt sensors. *Isokinet. Exerc. Sci.* **2021**, *29*, 175–184. [CrossRef]
30. Procházka, A.; Vyšata, O.; Mařík, V. Integrating the Role of Computational Intelligence and Digital Signal Processing in Education. *IEEE Signal Process. Mag.* **2021**, *38*, 154–162. [CrossRef]
31. Procházka, A.; Dostál, O.; Cejnar, P.; Mohamed, H.; Pavelek, Z.; Vališ, M.; Vyšata, O. Deep Learning for Accelerometric Data Assessment and Ataxic Gait Monitoring. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2021**, *29*, 33434133. [CrossRef] [PubMed]
32. Brennan, L.; Bevilacqua, A.; Kechadi, T.; Caulfield, B. Segmentation of shoulder rehabilitation exercises for single and multiple inertial sensor systems. *J. Rehabil. Assist. Technol. Eng.* **2020**, *7*, 2055668320915377. [CrossRef] [PubMed]
33. Alfakir, A.; Arrowsmith, C.; Burns, D.; Razmjou, H.; Hardisty, M.; Whyne, C. Detection of Low Back Physiotherapy Exercises With Inertial Sensors and Machine Learning: Algorithm Development and Validation. *JMIR Rehabil. Assist. Technol.* **2022**, *9*, e38689. [CrossRef]
34. Prochazka, A.; Schatz, M.; Tupa, O.; Yadollahi, M.; Vysata, O.; Valis, M. The MS Kinect Image and Depth Sensors Use for Gait Features Detection. In Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 2271–2274.
35. Shah, N.; Aleong, R.; So, I. Novel Use of a Smartphone to Measure Standing Balance. *JMIR Rehabil. Assist. Technol.* **2016**, *3*, e4. [CrossRef]
36. Skovbjerg, F.; Honoré, H.; Mechlenburg, I.; Lipperts, M.; Gade, R.; Naess-Schmidt, E. Monitoring Physical Behavior in Rehabilitation Using a Machine Learning–Based Algorithm for Thigh-Mounted Accelerometers: Development and Validation Study. *JMIR Bioinform. Biotechnol.* **2022**, *3*, e38512. [CrossRef]
37. Gu, C.; Lin, W.; He, X.; Zhang, L.; Zhang, M. IMU-based motion capture system for rehabilitation applications: A systematic review. *Biomim. Intell. Robot.* **2023**, *3*, 100097. [CrossRef]
38. Janáková, D. Foot and Ankle Kinematics in Patients with Femoroacetabular Impingement Syndrome. Master's Thesis, Charles University, Prague, Czech Republic, 2021.
39. Wouters, D.; Cavallaro, G.; Jensen, K.; East, B.; Jíšová, B.; Jorgensen, L.; López-Cano, M.; Rodrigues-Gonçalves, V.; Stabilini, C.; Berrevoet, F. The European Hernia Society Prehabilitation Project: A Systematic Review of Intra-Operative Prevention Strategies for Surgical Site Occurrences in Ventral Hernia Surgery. *Front. Surg.* **2022**, *13*, 847279. [CrossRef]
40. Ciomperlik, H.; Dhanani, N.; Cassata, N.; Mohr, C.; Bernardi, K.; Holihan, J.; Lyons, N.; Olavarria, O.; Ko, T.; Liang, M. Patient quality of life before and after ventral hernia repair. *Surgery* **2020**, *169*, 1158–1163. [CrossRef]
41. See, C.; Kim, T.; Zhu, D. Hernia Mesh and Hernia Repair: A Review. *Eng. Regen.* **2020**, *1*, 19–33.
42. Qabbani, A.; Aboumarzouk, O.; El Bakry, T.; Al-Ansari, A.; Elakkad, M. Robotic inguinal hernia repair: Systematic review and meta-analysis. *ANZ J. Surg.* **2021**, *91*, 2277–2287. [CrossRef]
43. Boukili, I.; Flaris, A.; Mercier, F.; Cotte, E.; Kepenekian, V.; Vaudoyer, D.; Passot, G. Prehabilitation before major abdominal surgery: Evaluation of the impact of a perioperative clinical pathway, a pilot study. *Scand. J. Surg.* **2022**, *111*, 14574969221083394. [CrossRef] [PubMed]
44. Cvetkovic, B.; Szeklicki, R.; Janko, V.; Lutomski, P.; Luštrek, M. Real-time activity monitoring with a wristband and a smartphone. *Inf. Fusion* **2018**, *43*, 77–93. [CrossRef]
45. Heredia-Elvar, J.; Juan-Recio, C.; Prat-Luri, A.; Barbado, D.; Vera-Garcia, F. Observational Screening Guidelines and Smartphone Accelerometer Thresholds to Establish the Intensity of Some of the Most Popular Core Stability Exercises. *Front. Physiol.* **2021**, *12*, 751569. [CrossRef]
46. Procházka, A. Rehabilitation Exercises and Computational Intelligence. Dataset, IEEE DataPort. 2024. Available online: https://ieee-dataport.org/documents/rehabilitation-exercises-and-computational-intelligence (accessed on 4 August 2024).
47. Dostál, O.; Procházka, A.; Vyšata, O.; Ťupa, O.; Cejnar, P.; Vališ, M. Recognition of Motion Patterns Using Accelerometers for Ataxic Gait Assessment. *Neural Comput. Appl.* **2021**, *33*, 2207–2215. [CrossRef]
48. Procházka, A.; Vyšata, O.; Ťupa, O.; Mareš, J.; Vališ, M. Discrimination of Axonal Neuropathy Using Sensitivity and Specificity Statistical Measures. *Neural Comput. Appl.* **2014**, *25*, 1349–1358. [CrossRef]
49. Martynek, D. Analysis of Rehabilitation Exercises Using Mobile Sensors. Mgr Thesis, University of Chemistry and Technology, Prague, Czech Republic, 2024.
50. Martynek, D. Rehabilitation Data Analysis and Processing. WWW Page, University of Chemistry and Technology, Prague, Czech Republic. 2024. Available online: https://danielmartynekdp.pythonanywhere.com/ (accessed on 4 August 2024).

51. Procházka, A.; Vyšata, O.; Vališ, M.; Ťupa, O.; Schatz, M.; Mařík, V. Bayesian classification and analysis of gait disorders using image and depth sensors of Microsoft Kinect. *Digit. Signal Prog.* **2015**, *47*, 169–177. [CrossRef]
52. Magris, M.; Iosifidis, A. Bayesian learning for neural networks: An algorithmic survey. *Artif. Intell. Rev.* **2023**, *56*, 11773–11823. [CrossRef]
53. Goh, E.; Ali, T. Robotic surgery: An evolution in practice. *J. Surg. Protoc. Res. Methodol.* **2022**, *2022*, snac003. [CrossRef]

sensors

Article

A Field-Programmable Gate Array-Based Adaptive Sleep Posture Analysis Accelerator for Real-Time Monitoring

Mangali Sravanthi [1,2], Sravan Kumar Gunturi [1], Mangali Chinna Chinnaiah [3,4,*], Siew-Kei Lam [4], G. Divya Vani [3], Mudasar Basha [3], Narambhatla Janardhan [5], Dodde Hari Krishna [3] and Sanjay Dubey [3]

1 Department of Electronics and Communication Engineering, Koneru Lakshmaiah Education Foundation, Aziz Nagar, Hyderabad 500075, Telangana, India; sravanthi.engg@mriet.ac.in or sravanthi.engg@gmail.com (M.S.); sravankumar.gunturi@gmail.com (S.K.G.)
2 Department of Electronics and Communication Engineering, Malla Reddy Institute of Engineering and Technology, Maisammaguda, Hyderabad 500014, Telangana, India
3 Department of Electronics and Communications Engineering, B. V. Raju Institute of Technology, Medak (Dist), Narsapur 502313, Telangana, India; divyavani.g@bvrit.ac.in (G.D.V.); mudasar.basha@bvrit.ac.in (M.B.); harikrishna.dodde@bvrit.ac.in (D.H.K.); sanjay.dubey@bvrit.ac.in (S.D.)
4 College of Computing and Data Science (CCDS), Nanyang Technological University, Singapore 639798, Singapore; siewkei_lam@pmail.ntu.edu.sg
5 Department of Mechanical Engineering, Chaitanya Bharati Institute of Technology, Gandipet, Hyderabad 500075, Telangana, India; njanardhan_mech@cbit.ac.in
* Correspondence: chinnaaiah.mc@bvrit.ac.in or chinnaiah.mangali@ntu.edu.sg

Abstract: This research presents a sleep posture monitoring system designed to assist the elderly and patient attendees. Monitoring sleep posture in real time is challenging, and this approach introduces hardware-based edge computation methods. Initially, we detected the postures using minimally optimized sensing modules and fusion techniques. This was achieved based on subject (human) data at standard and adaptive levels using posture-learning processing elements (PEs). Intermittent posture evaluation was performed with respect to static and adaptive PEs. The final stage was accomplished using the learned subject posture data versus the real-time posture data using posture classification. An FPGA-based Hierarchical Binary Classifier (HBC) algorithm was developed to learn and evaluate sleep posture in real time. The IoT and display devices were used to communicate the monitored posture to attendant/support services. Posture learning and analysis were developed using customized, reconfigurable VLSI architectures for sensor fusion, control, and communication modules in static and adaptive scenarios. The proposed algorithms were coded in Verilog HDL, simulated, and synthesized using VIVADO 2017.3. A Zed Board-based field-programmable gate array (FPGA) Xilinx board was used for experimental validation.

Keywords: sleep posture recognition; adaptive posture analysis; FPGA; sensor fusion

Citation: Sravanthi, M.; Gunturi, S.K.; Chinnaiah, M.C.; Lam, S.-K.; Divya Vani, G.; Basha, M.; Janardhan, N.; Hari Krishna, D.; Dubey, S. A Field-Programmable Gate Array-Based Adaptive Sleep Posture Analysis Accelerator for Real-Time Monitoring. *Sensors* **2024**, *24*, 7104. https://doi.org/10.3390/s24227104

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 14 September 2024
Revised: 31 October 2024
Accepted: 1 November 2024
Published: 5 November 2024

1. Introduction

The human life cycle and health are linked to sleep duration and posture. Over the last three decades, researchers have conducted sleep analysis and monitoring. According to medical reports and history, the impacts of improper sleep on daily life include musculoskeletal strain, respiratory issues, circulation problems, reduced sleep quality, and digestive issues. Indirect poor sleep affects human behavior and daily activities. Research has shown that 9% to 38% of the general population is affected by sleep apnea [1]; in the future, this is expected to increase. Sleep posture recognition and analysis are crucial for researchers and medical systems when recommending various medications and other equivalent systems for better sleep in patients as well as the general population. The American Sleep Disorders Association and Sleep Research Society have been investigating the impact of sleep disorders on human activities for the last four decades [2,3].

To date, sleep posture analysis has faced various challenges, including data acquisition, in assisting patients and individuals. Data acquisition has been performed by researchers using wearable and non-wearable pressure and non-contact sensing devices [4,5]. Piezoresistive arrays or pressure-sensor-based bedsheets and beds are utilized for data acquisition for posture analysis [5,6]. A few challenges are raised with these methods; the data acquisition error percentage is higher in this case, and a large amount of data are required to compute the posture. Other researchers have conducted posture analysis using wearable devices; this approach has been used in initial learning and regular patient monitoring [7,8]. Recent advancements in sleep posture data acquisition have been achieved using non-contact methods, such as radar or ultrasonic sensor arrays [5] and vision approaches. Wearable sensors such as 3-axis accelerometers [9] and thermostats, as well as electromyography (EMG), electroencephalography (EEG), and photoplethysmography (PPG) devices [10], have been utilized for sleep posture data capture. These wearable sensors are integrated into Fit-Bit modules or smart watches. Similarly, non-wearable technology and sensors, such as multimodal sensor fusion [11] and smart phones [12], have been utilized for sleep posture analysis by various researchers. Selection of the sensor type is a challenge in sleep posture analysis.

Different sensor data have been processed using data processing analysis techniques, such as data mining and classifiers. Computation methods play a vital role in sleep posture data analysis. Classifiers such as random forests and binary-type decision trees, as well as supervised classifiers such as hidden Markov models (HMMs), support vector machines (SVMs), and k-nearest neighbors (kNNs) [13], have been employed. Classifiers are used for learning, and real-time feature matching can be utilized to determine sleep posture. In this regard, researcher-driven deep learning methods have been adopted in sleep research, such as recurrent NNs (RNNs), long short-term memory (LSTM) networks [6], convolutional NNs (CNNs), and generative adversarial NNs (GNNs) [14]. In this process, computing devices such as microcontrollers, GPUs, and CPUs, as well as cloud computing, are essential for sleep posture data acquisition, data processing with classifiers, and posture accomplishment [15,16]. Research studies addressing sleep posture require low-power-consuming devices and effective computation for analysis during learning and real-time implementation in static and adaptive states. Edge computing devices, such as FPGAs, are essential for real-time sleep posture systems at present and in the future, making it challenging to provide complete solutions.

The proposed hardware-efficient methods provide sleep posture analysis for present and future usage in real-time implementations. The FPGA-based adaptive sleep posture analysis accelerator has three novel embedded methods:

1. An FPGA-based learning algorithm for acquiring data regarding the standard and adaptive conditions of sleep postures. Real-time adaptive learning was developed for various subjects (humans).
2. An FPGA-based hierarchical binary classifier (HBC) algorithm was developed for the classification of sensor fusion data in the learning and analysis stages for event-driven conditions.
3. Hardware-based sleep posture analysis is the next stage of the proposed method. The FPGA-based solution for sleep posture analysis is the first of its kind for adaptive-based event conditions.

This paper is structured as follows. This section presents the background and motivation for sleep posture analysis research using FPGA implementation. In Section 2, the details of the proposed methodology are presented with theoretical and hardware schemes. The proposed method was validated, and the results are presented in Section 3 in the form of synthesis, power consumption, and experimental details with comparison. The final section concludes the study with future perspectives.

2. Hardware-Based Algorithms

Sleep posture analysis was executed in two stages as per the proposed method. Hardware-based algorithms were first developed for sleep posture analysis, and secondly, hardware schemes for analysis were explored. See Table 1.

Table 1. Proposed research-related abbreviations.

Abbreviation	Definition
S_{PR}	S: Ultrasonic sensors {H_R, H_L, A_R, A_L, R_L and L_L} P: Position of sensor at head (H), abdomen (A), and limb (L) R: Position at right (R) and left (L) sides
PP	Past posture
CP	Current posture
T	Time
Postures	Right yearner (RY), left yearner (LY), left fetal (LF), right fetal (RF), left lateral posture (LLP), and supine posture (SP)
LUT	Look-up table
P_C	Pose current
Pp	Pose past
Tp_C	Pose T current
Tpp	Pose T past
Ps	Pose Static

2.1. Hardware-Based Algorithm for Sleep Posture Analysis

Figure 1 presents an overall flowchart of the proposed hardware-based sleep analysis. The initial conditions are as follows: The algorithm checks whether there is a subject on the bed. Next, it determines whether data have previously been recorded for this subject's profile. If the subject is new to the sensor radar, it starts learning all their postures; such an adaptive process not only relies on near-sensor fusion data. If the subject is recognized, their posture is classified using a hardware-based hierarchical binary classifier (HBC) algorithm. The HBC provides the along class, and the subject is either in a static or adaptive state. In a static pose, the subject's posture is recorded. If the subject switches from one pose to another, this is recorded along with the time spent in the respective postures.

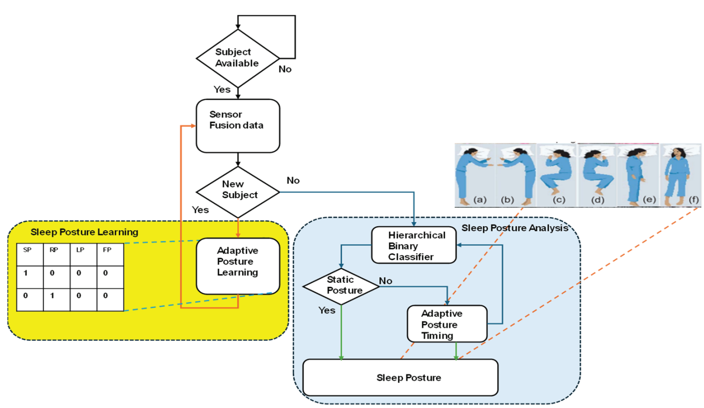


Figure 1. Flowchart of proposed hardware-based sleep posture analysis. (**a**) right yearner (RY), (**b**) left yearner (LY), (**c**) left fetal (LF), (**d**) right fetal (RF), (**e**) left lateral posture (LLP), and (**f**) supine posture (SP).

The sleep posture details are presented in Figure 2 and are broadly classified into four groups: supine (SP), left (LP), right (RP), and frog posture (FP). Posture details are learned by an edge computing device using sensor fusion data. These posture data are learned at the sleep-posture-learning stage and are utilized in the sleep posture analysis in the classifier stage, defining the type of posture.

(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 2. (**a**) right yearner (RY), (**b**) left yearner (LY), (**c**) left fetal (LF), (**d**) right fetal (RF), (**e**) left lateral posture (LLP), and (**f**) supine posture (SP) [17].

2.1.1. Hardware-Based Algorithm for Sleep Posture Learning

This section addresses sleep posture learning. Algorithm 1 represents the pseudocode of the sleep posture learning in versatile scenarios.

Algorithm 1: Pseudocode for hardware-based sleep posture learning

```
1.  Initialize sensory distance into sensor fusion data (SF)
2.  always @ (posedge clk) begin
3.   {sleep_posture_analysis, learn_new_posture, adapt_sf_data, retry} = 0.
4.  Case (state)
5.  INIT: Next State = WAIT_PIR.
6.  WAIT_PIR: Next_State = (PIR == 1)? CHECK_DATA_RANGE: WAIT_PIR.
7.  CHECK_DATA_RANGE:
8.  Next_state= (subject data == range data)? CHECK_POSE: ADAPT
9.  CHECK_POSTURE:
10.    next_state = (posture_HBC)? ANALYSIS: Learn.
11. ANALYSIS: {sleep_posture_analysis = 1, next_state = WAIT_PIR};
12. LEARN: {learn_new_posture = 1, next_state = WAIT_PIR};
13. ADAPT: {adapt_sf_data = 1, next_state = WAIT_PIR};
14. RETRY: next_state = (PIR == 1)? CHECK_DATA_RANGE: RETRY.
15. default: next_state = INIT.
16.     end case, End.
```

Algorithm 1 describes the pseudocode for hardware-based sleep posture learning. In step 1 (line 1), the sensors are initialized; the data in the sensor fusion from the sleep posture details are presented in Figure 2. The flowchart of Algorithm 1 is shown in Figure 3. The system clock synchronizes the sensor fusion data and enables the sleep posture to be learned with a continuous, adaptive approach (lines 2 and 3). The PIR sensor data are utilized to determine whether there is a human on the bed (line 6). Sensor fusion data are classified into two forms, standard posture and time-invariant data, which are considered adaptive sleep postures (line 8). Standard posture information related to existing subject postures is then determined (line 10). The proposed method learns any new postures and registers them using the hierarchical binary classifier (HBC) (line 12). Sleep postures are analyzed using the HBC method (line 11). The subject's movement from one posture to another is evaluated as an adaptive posture (line 13). Adaptive posture evaluation is essential for assisting the subject in their struggle with pain or other issues as it alerts attendants to the medical assistance system. This is a continuous process in the estimation of sleep postures (lines 14 and 15).

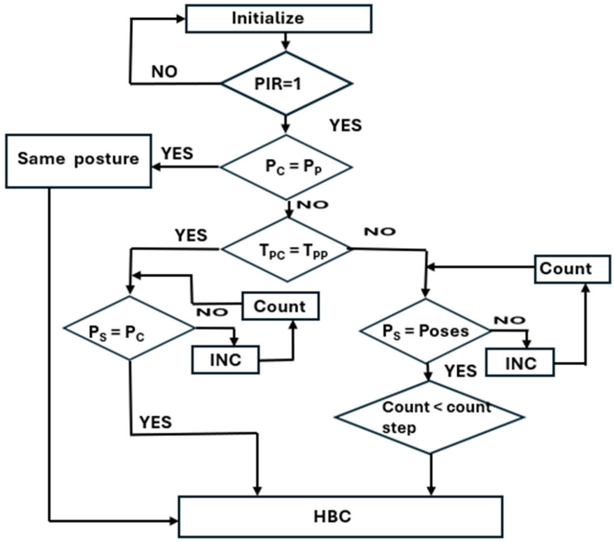


Figure 3. Flowchart for sleep posture learning.

2.1.2. Hardware-Based Hierarchical Binary Classifier Algorithm for Sleep Posture

This subsection presents the sensor fusion data classified using the hierarchical binary classifier (HBC) algorithm. Each sleep posture is learned as described in Section 2.1.1 and classified with respect to sleep posture, as shown below.

Figure 4 presents the sensor fusion data captured for the left lateral posture (LLP) and supine posture (SP). Table 2 lists the details of the sensor fusion data for sleep posture analysis. Figure 4 and Table 2 provide basic information for the execution of the hierarchical binary classifier (HBC) algorithm.

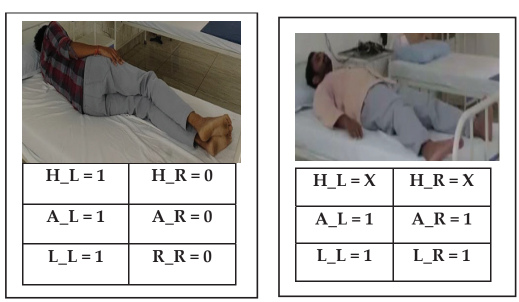


Figure 4. Left lateral posture (LLP) and supine posture (SP) [17] with sensor fusion data.

Table 2. Sensor fusion data for sleep posture analysis.

Posture	**Head_Right (H_R)**	**Head_Left (H_L)**	**Abdomen_Right (A_R)**	**Abdomen_Left (A_L)**	**Right Leg (R_L)**	**Left Leg (L_L)**
Right yearner (RY)	1	1	0	1	0	1
Left yearner (LY)	1	1	1	0	1	0
Left fetal (LF)	1	1	1	1	0	1
Right fetal (RF)	1	1	1	1	1	0
Right lateral posture (RLP)	1	0	1	0	1	0
Left lateral posture (LLP)	0	1	0	1	0	1
Supine posture (SP)	X	X	1	1	1	1
Frog posture (FP)	1	1	1	1	0	0

Algorithm 2 and the flow chart in Figure 5 provide details of the hardware-based HBC algorithm for sleep posture learning and analysis. Data from the six pairs of sensors are captured from the positions of the human limbs, abdomen, and head. Sensor fusion data initialization and triggered operations are performed in lines 1 and 2, respectively. The HBC classifies data into three positions: lower limb, abdomen, and head (lines 3–14).

Algorithm 2: Pseudocode for hardware-based hierarchical binary classifier algorithm

```
1.   Initialize sensory distance into sensor fusion data (SF)
2.   always @ (posedge clk) begin {H_R, H_L, A_R, A_L, R_L & L_L} = 0;
3.   Case (Posture Data)
4.    Posture: (R_L & L_L =2′b00)? Frog posture: Posture _1.
5.    Posture _1: (R_L & L_L =2′b11)? Supine posture: Posture _2.
6.    Posture _2: (R_L & L_L =2′b01)? Right posture: Left posture.
7.    Case (Right posture)
8.    Right_1: (A_R & A_L=2′b11)? Right Foetus: Right_2.
9.    Right_2: ((A_R & A_L=2′b01) && (H_R & H_L=2′b11))? LY: RLP.
10.    Case (Left posture)
11.  Left_1: (A_R & A_L=2′b11)? Left Foetus: Left_2.
12.  Left_2: ((A_R & A_L=2′b10) && (H_R & H_L=2′b11))? RY: LLP.
13.  Default {H_R, H_L, A_R, A_L, R_L & L_L} = 0.
end case, End.
```

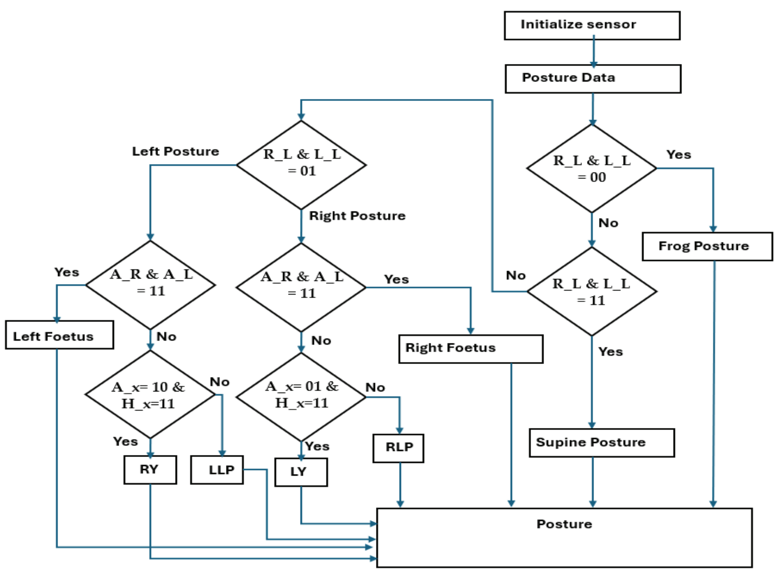


Figure 5. Flow chart of hierarchical binary classifier algorithm for sleep posture.

Early classification is performed using the lower limb position when the right and left lower limbs (R_L and L_L) are equal to the two-bit information 2′b00. When the limbs are detected as not positioned at the lower end of the bed, the subject's pose is determined to be a frog posture (line 4). If this condition fails, then the classifier switches to the next state. When the lower limbs are 2′b11, the supine posture is recorded (line 5); otherwise, the process switches to the next state. The classifier utilizes sensor data for the entire lower limbs to establish the right (lines 7 to 9) or left (lines 10 to 12) side postures. When the abdomen sensor data (A_R and A_L) are equal to 2′b11, they are recorded as the right fetal position (line 8), similar to the left fetal position mentioned in line 11. When lower limb and abdomen sensor fusion data cannot be classified, the HBC depends on the head position on the right or left side (H_R and H_L). The final stage of the hierarchy classifies the postures of the left yearner (LY) and right lateral posture (RLP) (line 9), and similar line postures are classified as the right yearner (RY) or left lateral posture (RLP) (line 12) based on the hierarchy conditions. Default conditions are considered to indicate that the postures are not available or improper (line 13). The adaptive approach considers improper posture evaluation for learning and analysis using time-of-flight (ToF) methods.

2.1.3. Hardware-Based Adaptive Sleep Posture Analysis Algorithm

The proposed hardware-based adaptive algorithm for sleep posture analysis under time-varying event conditions is presented in this subsection. Adaptive sleep posture analysis provides accurate information regarding the status of the patients/elderly individual's motion and sleep conditions. This method depends on the selection of the sensor, which provides the time-of-flight-based sensing, time-stamped data, and time-dependent features and drives the detection of human motion on the bed. Few researchers have used hidden Markov models for motion detection [18,19]. The proposed hardware-based algorithm was developed using a binary-based adaptive threshold for digitizing the motion action in sleep postures. A human transitioning from a left lateral posture (LLP) to a supine posture (SP) is shown in Figure 3.

The sensor captures and calculates the distance to the object or human based on the echo signal between the sensor and human. The formula used is as follows:

$$\text{Distance} = \frac{\text{Speed of Signal} \times \text{Time of Flight}}{2} \tag{1}$$

Algorithm 3 and Figure 6 present the adaptive sleep posture evaluation method. In this algorithm, the time-variant data are the main concern with respect to posture. This algorithm operates concurrently to evaluate the timing of static and dynamic human postures with respect to other postures. Once the human subject is identified through the PIR sensor using sensor fusion data, their pose is determined. During sleep, patients in pain change from one posture to another. In this context, postural dynamics are essential for estimating patient challenges. Until the pose changes, it is considered to be the same posture as that mentioned in line 4. When posture changes are not observed, the time duration is evaluated using the counter count_1 (lines 9 to 10). If the pose is associated with a time variable, it is evaluated in lines 6–8. The count parameter is deployed to calculate the timing of the dynamic posture variants (line 7). The adaptive posture is defined as a two-level posture, and the process reverts to the original posture, identified as per line 1. Otherwise, the posture is evaluated using the HBC, and the learning time is recorded. This timing is useful, and abrupt changes in the posture time will improve the adaptiveness of the method.

Algorithm 3: Pseudocode of adaptive-based algorithm for sleep posture analysis

```
1.   Sensor fusion data & Present posture, count_1 = 8′d0, count = 8′d0.
2.   If (PIR)
3.   Case (Pose)
4.   Pose 1: (Pose current = Pose past)? Same posture: Pose 2.
5.   Pose 2: (Pose_T_current = Pose_T_past)? Pose 6: Pose 3.
6.    Case (Pose 3)
7.    Pose 4: (Pose_ static = Poses)? count: count <= count + 1.
8.    Pose 5: (count > count_step1)? HBC: Pose 1.
9.    Case (Pose 6)
10.   Pose 7: (Pose_ static = Pose current)? count_1: count_1 <= count_1 + 1.
11.   end case, End.
```

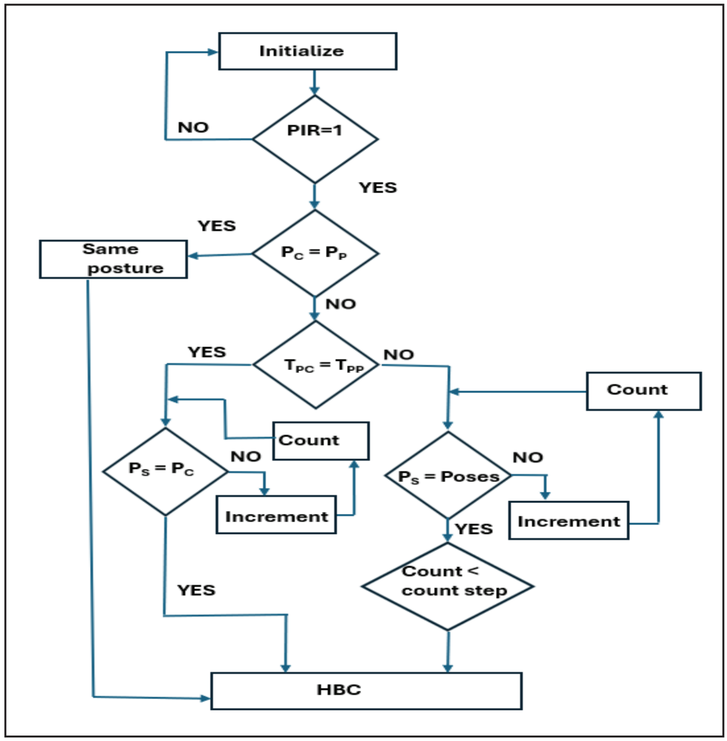


Figure 6. Flow chart of adaptive-based algorithm for sleep posture analysis.

3. Hardware Schemes for Sleep Posture Analysis

This section outlines hardware schemes that are equivalent to the hardware-based algorithm proposed in Section 2. The hardware schemes for the sleep posture analysis are given in the following. Initially, the hardware accelerator, learning, HBC, and adaptive posture evaluation-based hardware schemes are presented.

3.1. Hardware Accelerator for Sleep Posture Analysis

Figure 7 presents the hardware accelerator for the analysis at the learning and evaluation stages. The hardware accelerator is integrated with sensors as represented in orange colours, a Wi-Fi module, and a display unit is represented with red colour line in the figure. the data bus is represented with dark and light blue colour. The overall architecture is controlled using the proposed algorithm with a control unit as represented with yellow colour data bus. The control unit operates and synchronizes the overall system. Passive infrared motion (PIR) sensors enable the system to capture the next level of data. In the absence of a human on the bed, PIR provides logic '0', and the accelerator continues in the sleep condition until PIR = '1'. Six pairs of ultrasonic sensors were integrated to capture sleep data. The sensors were concurrently triggered by a control unit. Every 20 bits of sensory data are shared with the distance converter and the fusion module. The control unit initially shares the data with the static posture-learning PE module; if any time-variant data are observed, it switches to an adaptive posture-learning module using a demultiplexer (DEMUX). Posture learning is performed by comparing the sensor fusion data with the reference subject data.

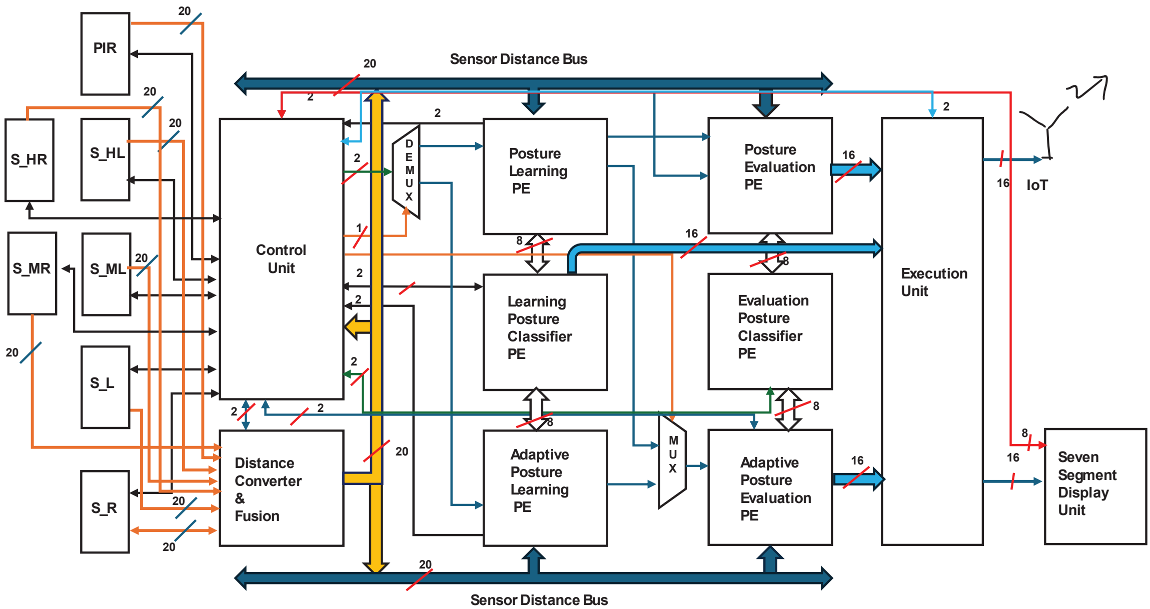


Figure 7. Overall hardware accelerator for sleep posture analysis.

The learning posture utilizes an HBC PE and communicates using 8-bit data. Postures that are not part of the reference model or the time-variant-type posture are learned using the adaptive posture-learning PE. If the pose is within the range of the reference poses, the learning stage enables the posture evaluation of the PE. The posture evaluation PE interfaces with the posture HB classifier module for evaluation. The defined posture is processed as an output through the execution unit (EU). The output is presented in the form of messages to the attendants/service team through the Wi-Fi module. These messages are displayed on the FPGA-based, seven-segment LCD display. This novel approach is used to adaptively analyze variations in the pose of the patient/elderly subject. In the learning stage, if the posture is not a standard pose, it is recorded as an adaptive posture of the subject. Under certain conditions, the subject performs the posture with respect to the time variable, and the adaptive posture evaluation PE registers the posture and compares it to the time-variant data between postures, and this is fed to the output.

3.2. Hardware Schemes for Sleep-Posture-Based Learning

The proposed equivalent internal architecture of Algorithm 1, referred to as the integration of the generic and adaptive posture-learning PEs, is presented in Figure 8. The architecture, enabled by the PIR sensor and ultrasonic sensor fusion distance data, has a FIFO memory structure. Prior postures are analyzed and estimated via either generic posture learning or adaptive learning using the posture-matching module. The posture-matching module was developed with three stages of FIFO, which are stored in FIFO_N-2, FIFO_N-1, and FIFO_N. Each FIFO dimension is an array of six pairs of sensors {H_R, H_L, A_R, A_L, R_L, and L_L}, each 20 bits in size. The shift encoder shifts the FIFO data from N-2 to N and performs in line with the concurrent barrel-shifter-based approach.

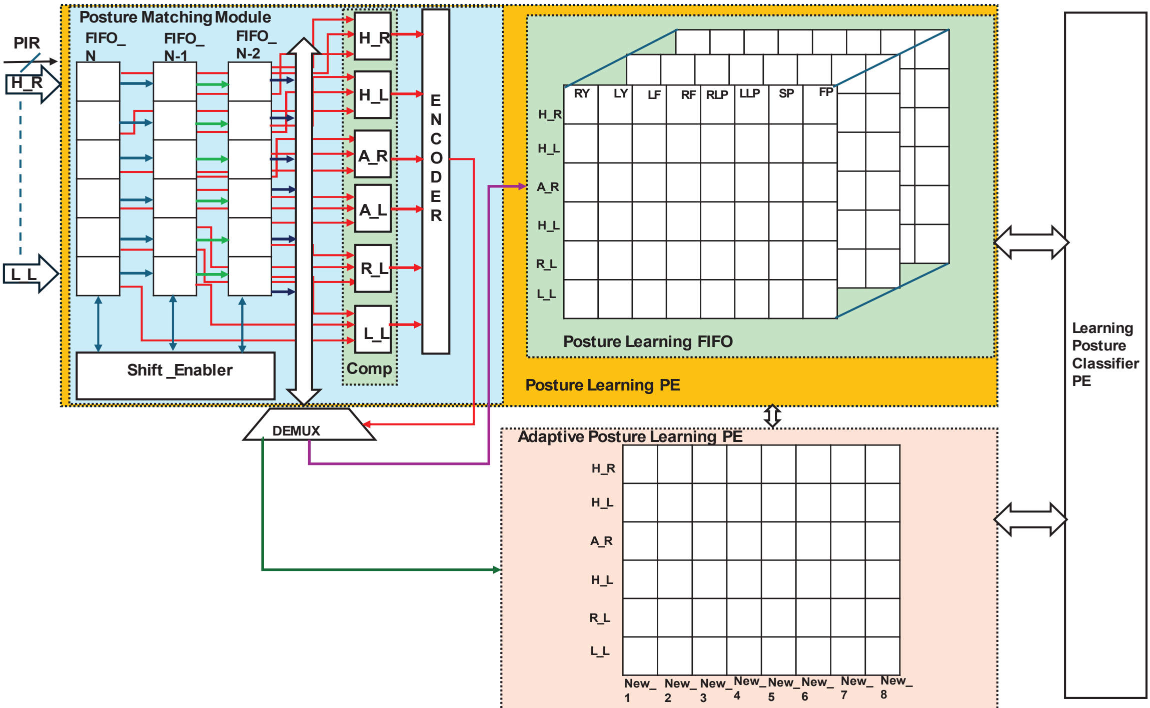


Figure 8. Internal architecture of sleep-posture-based learning.

The FIFO data are stored for every 1/6 s and compared using the 'comp' module. L_L sensor-based arrays of N-2, N-1, and N are compared at the L_L comparator at the same distance. H_R to L_L are encoded, and when 6′b11111 is presented in three iterations, new generic learning is enabled in the posture-learning PE. Otherwise, adaptive posture learning is enabled in the PE. The posture-learning PE consists of an internal control module and FIFO posture learning. The internal control shares the present status with the control unit, as presented in Figure 4 with different colours. FIFO posture learning is performed for the standard postures of each subject, as presented in Table 2, from right yearner (RY) to the frog posture (FP). Eight standard postures are included. A maximum of eight subjects are recorded, and the learning methods replace unused subject data with new subject data with control unit permissions. The authors attempted to optimize memory using time-variant-based learning. This method is optimized for storing datasets, as the execution of the real-time interference stage faces memory issues. In this regard, time-variant-based learning is called adaptive learning, which is employed in two scenarios: past subjects with a new posture and new subject postures. The adaptive posture-learning PE is activated as per the posture-matching module; it stores New_1 to New_8. H_R to L_L sensor data are stored as new, standard subject postures. It is also able to memorize new postures for a new subject, which are labeled in the posture-learning PE. A known subject with a new posture is allocated a new label beside the FP posture through the FIFO posture learning. This was a novel attempt, at the inference level, to avoid dataset memory issues. Both learning methods are regularly interfaced with the learning posture classifier PE for posture estimation, as shown in Figure 9.

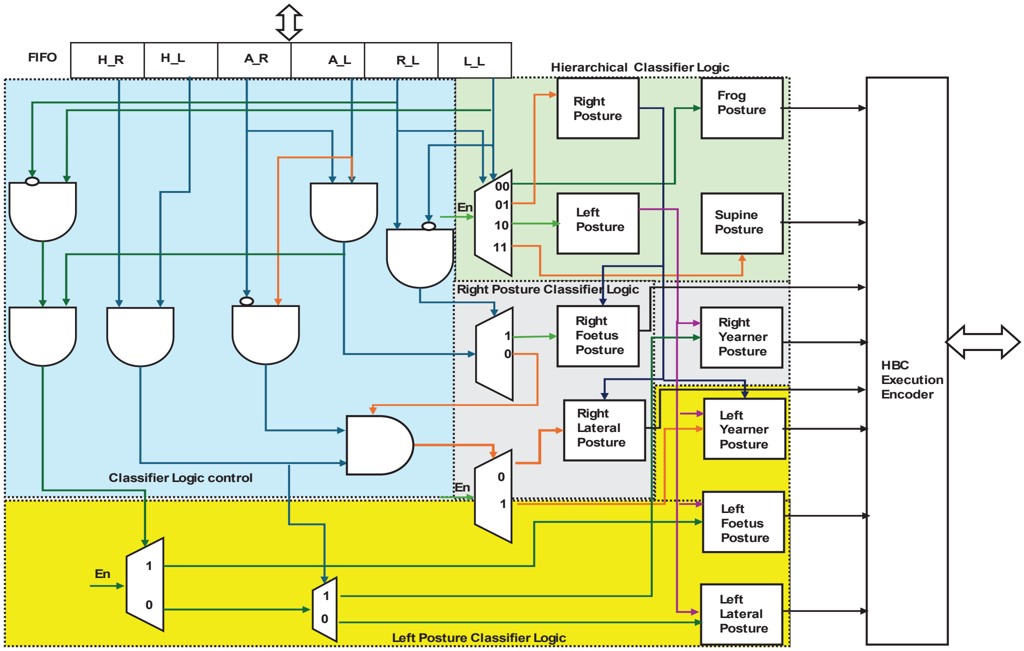


Figure 9. Hardware scheme for sleep-posture-based hierarchical binary classifier (HBC) PE.

3.3. Hardware Schemes of Hierarchical Binary Classifier for Sleep Posture

Figure 9 presents the internal hardware schemes of the HBC for sleep posture and signal information is represented in different colours. The proposed binary classifier was developed in line with the [20,21] binary search tree. The binary search tree classifier was organized with the center weights of the tree; in the proposed approach, it is heuristic, with a hierarchical binary classifier.

The HBC is embedded with a 20-bit FIFO structure, (sensor distance) × 6 (H_R, H_L, A_R, A_L, R_L and L_L) × 1, and receives the data from the learning or posture evaluation modules, the hierarchical classifier logic, classifier logic control, right posture classifier logic, and left posture classifier logic. As per the binary search tree, R_L and L_L enable 2-bit information to act as a selector in the hierarchical classifier logic. The frog posture (FP) and supine posture (SP) were selected as 2′b00 and 2′b11 and 2′b01 and 2′b10 to enable the right and left posture classifiers. As per the binary hierarchy, with A_R and A_L data, the classifier logic control defines the right and left postures in detail. A_R and A_L as 2′b11 and R_L and L_L as 2′b10 define the right fetal (RF) postures and enable the selection of the right lateral posture (RLP). The next level of hierarchy is defined with the top H_R and H_L bits for the selection of the left postures, such as the lateral, fetal, and yearner postures. Overall, data from the six pairs of sensors (6′b111101) define the left fetal position using the left posture classifier logic. Similarly, lines 6′b010101 classify the left lateral posture (LLP) and 6′b111010 the left yearner (LY) posture. Adaptive learning regularly performs subclassification in the HBC according to new requirements. The posture modules are interfaced with the HBC encoder and with the posture evaluation and learning modules.

3.4. Hardware Schemes for Adaptive Posture Evaluation

Algorithm 3 presents hardware-based adaptive posture evaluation using a time-variant approach. The sensor's distance varies with time and is considered adaptive in sleep posture evaluation. Figure 10 presents the internal hardware schemes for adaptive posture evaluation with different colour lines for data information.

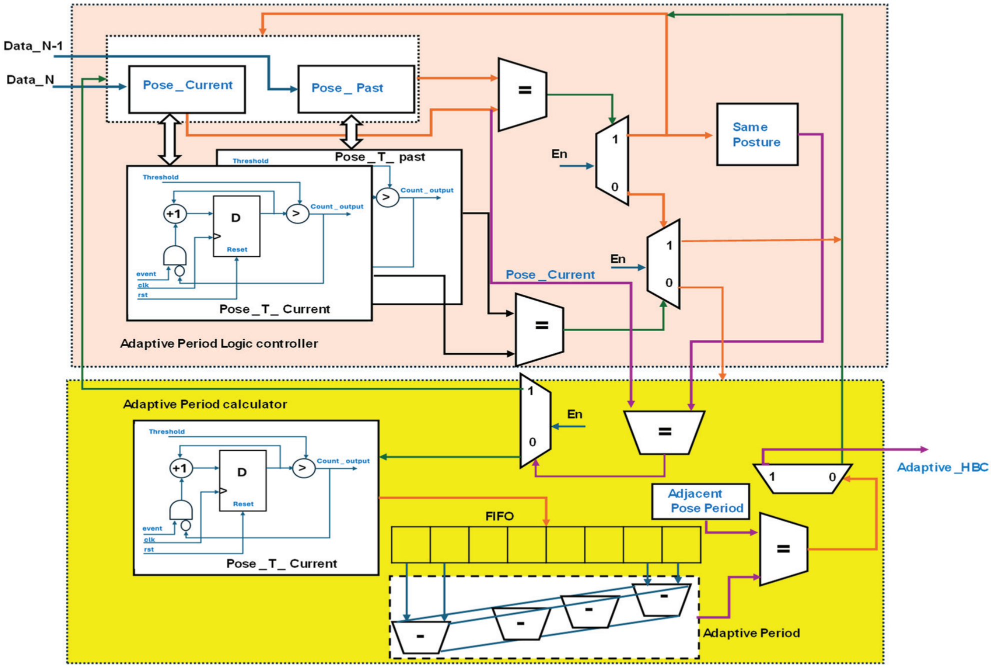


Figure 10. Hardware scheme for adaptive posture evaluation PE.

The adaptive posture evaluation is integrated with an adaptive period logic controller and adaptive period calculator. At the logic controller, data originating from the sensor distance bus are utilized to ascertain the past and current poses. When registered concurrently, once the pose is initiated, the respective counters are triggered until the pose changes. These counters operate with a device clock frequency of 100 MHz and are synchronized with the AXI lite protocol. The Pose _ T _ Current and Pose _ T _ past counters are counts utilized in the event of posture registration and reach the threshold for posture change. If both the count and pose are the same, the posture is recognized as the same as that classified with the HBC. Otherwise, the adaptive period calculator of time-variant sleep posture data is enabled. The Pose _ T _ Current module of the period calculator registers data for every 1/6 s and determines the difference using the 2's complement adder method. The duration from the current posture to the adjacent posture time is considered the adjacent pose period. Once both the adaptive and adjacent pose periods are equal, the HBC evaluates the posture-to-posture time. The adaptive period module iterates until the adjacent pose periods match.

4. Results

The proposed FPGA-based accelerator for determining sleep posture under generic and adaptive conditions is presented in this section. The results are compared in terms of FPGA resource utilization in implementation, along with power consumption based on the hardware schemes presented in Figures 7–10. The proposed approach was validated through real-time experiments with six ultrasonic sensor pairs and a ZedBoard family field-programmable gate array (FPGA).

4.1. Resource Utilization

The proposed approach is the first of its kind to use the FPGA-based accelerator for non-contact sleep posture analysis. The hardware schemes were coded using Verilog HDL, and the Xilinx simulator was used for their functional verification. Vivado tools 17.3 version was utilized for HDL synthesis for bit generation. Xilinx tools and the FPGA were procured from the Xilinx university program.

Xilinx (San Jose, CA, USA) produced the Xilinx Zynq XC7Z020-1CSG484 ZedBoard, which features approximately 85,000 programmable logic cells. The device incorporates look-up tables (LUTs) and flip-flops for executing logic operations and short-term memory storage. The board's BRAM, accessible via AXI lite, comprises over 140 blocks of 36 kb each (totaling 4.9 Mb), which are used to store sensor fusion and intermediate data. In the proposed design, BRAM is primarily utilized in FIFO. The board also includes about 220 DSP slices (18 × 25 MACCs) for handling data transfer and other computational tasks. Table 3 illustrates how these resources are employed in the proposed approach.

Table 3. ZedBoard FPGA resource utilization for sleep posture analysis accelerator.

Module	LUT	BRAM	DSP Slices
Interfacing modules (sensors, communication (UART), Xilinx IP cores)	6362	16	22
Sleep-posture-based learning PE	4404	12	15
Hierarchical binary classifier (HBC) PE	3916	8	12
Adaptive posture evaluation PE	5140	22	25
Control unit and PWDC sensor fusion	2692	9	12
Execution modules and display	1958	5	10
Total	24,472	72	96

Table 3 presents the device's utilization of the sleep posture analysis accelerator. FPGA-based accelerators provide fast computing and low power consumption [22–24]. The total device utilization for the proposed approach in the form of look-up tables (LUTs), block RAM (BRAM), and digital signal processing (DSP) slices was 46% (24,472), 51% (72), and 44% (96), respectively.

Figure 11 presents a quantitative analysis of the resources consumed by the device in the interfacing module, which are 26% for LUT, 11% for BRAM, and 10% for the DSP slices. Similarly, other modules include the sleep-posture-based-learning PE (18%, 9%, and 7%), hierarchical binary classifier (HBC) PE (16%, 6%, and 6%), adaptive posture evaluation PE (21%, 16%, and 11%), control unit and PWDC sensor fusion (11%, 6%, and 6%), and execution module and display (8%, 4%, and 4%). It is observed from device utilization that the interfacing modules and adaptive posture evaluation PE consume more resources. The interface is embedded with UART using AXI lite to establish and communicate sleep postures to other devices using Wi-Fi ESP8266 external devices. The resource consumption of BRAM is approximately 51%, and the sensor fusion distance is passed in a few stages through FIFO. AXI-based FIFO and an IP core are utilized as part of the programmable logic (PL) of the ZedBoard FPGA.

Figure 12 shows the power consumption of the device when computing a reconfigurable device (FPGA). Overall, the static device power consumption as per the Xilinx power estimator (XPE) is 1.2 watts. The PE evaluation of the adaptive posture consumed 32% of the overall power. The power consumption of other components was also obtained from the XPE analysis. Interfacing with external modules consumes the second-largest share of power.

The total power consumption is 1.2 watts, with the dynamic power and static power consumption comprising 0.96 watts and 0.24 watts. Hardware schemes were designed with eight pipeline stages (S), and a device clock time (Tclk) of 10 ns; a total of 30 (N) iterations were utilized for validation. The 370 ns latency of the proposed hardware schemes is represented in Equations (2) and (3).

$$\text{Latency per iteration} = 8 \times 10\ \text{ns} = 80\ \text{ns} \tag{2}$$

$$\text{Total latency} = (\text{N} + \text{S} - 1) \times \text{Tclk} = (30 + 8 - 1) \times 10\ \text{ns} = 370\ \text{ns}. \tag{3}$$

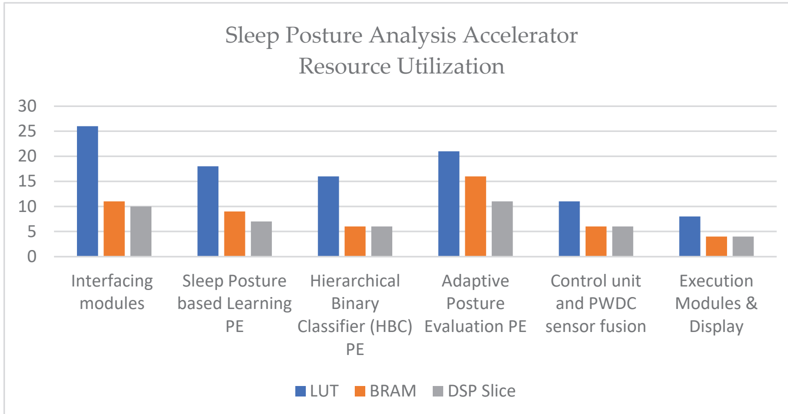


Figure 11. Resource utilization of sleep posture analysis accelerator.

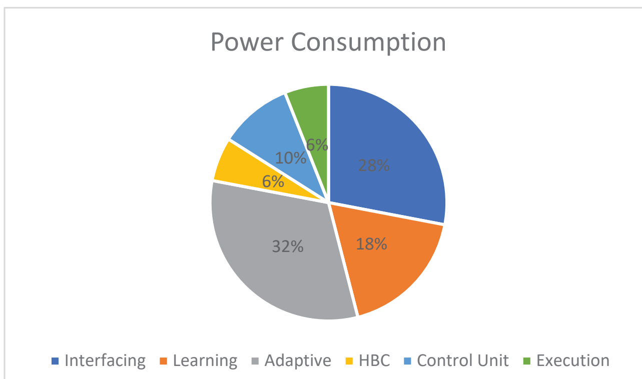


Figure 12. Device power consumption of sleep posture analysis accelerator.

The overall accuracy is 98.4%. This was computed based on the data captured from the multiple sensors and their fusion. The total number of correct data predictions were 30 and 29, respectively. Equations (4) and (5) represent the accuracy and error rate formulae. An accuracy of 98.4% and error rate of 1.6% are mentioned in Equations (6) and (7).

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Predictions}} \times 100 \tag{4}$$

$$\text{Error rate} = 1 - \text{Accuracy} \tag{5}$$

$$\text{Accuracy} = \frac{29}{30} \times 100 = 98.4\% \tag{6}$$

$$\text{Error rate} = 1 - \frac{29}{30} \times 100 = 1.6\% \tag{7}$$

4.2. Experimental Results

This section describes the experimental setup and experiments for the validation of the proposed sleep posture analysis accelerator.

4.2.1. Experimental Setup

Figure 10 illustrates sleep posture analysis using the contactless approach proposed in this study. As shown by the experimental setup in Figure 13, the side views of A_L and A_R are different for each subject. The results are shown in Figure 14. The experimental setup presents the integration of the six ultrasonic sensor pairs from rhydolabz that utilize an ultrasound echo signal. The sensors were operated with a 40 KHz frequency and voltage range of 3.3–5 V, and consumed a 5 mA current. The voltage was fetched from the voltage regulators of 7805 IC modules. Ultrasonic sensors were employed to detect objects using pulse-width modulation (PWM)-based echo signals that were digitized using an FPGA. The ultrasonic sensor is capable of capturing data within the ideal distance of 3 m; however, to remove redundancy and other noise, a 2.7 m to 0.3 m range was more suitable.

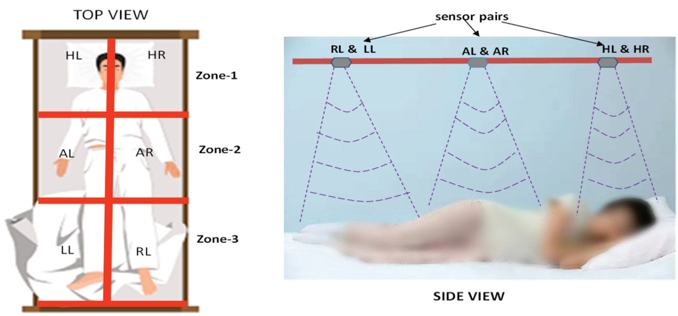


Figure 13. Illustration of experimental setup of contactless sleep posture analysis.

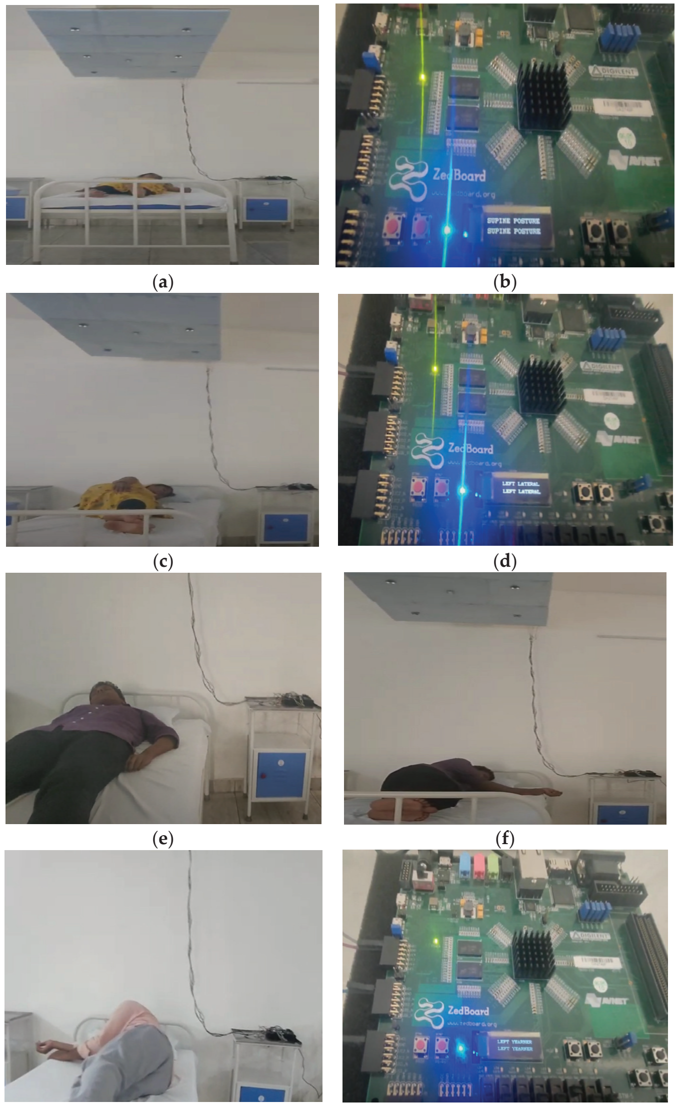
(**a**) without subject (**b**) without subject

Figure 14. Experimental setup of bed for sleep posture analysis without and with human.

The PIR sensor was positioned between the H_L and H_R sensors to estimate whether the bed was occupied by a human. The PIR sensor range covered around 3 m from its position. As shown in Figure 14, the FPGA device was placed on a bedside table and was used to compute the sleep posture as per the proposed approach. The bed size used in this experiment was approximately 110 cm. The sensor was positioned from the bed at a height of 2.4 m, and each sensor covered approximately 47 cm of the bed. The sensors were positioned to cover an inner-bed range of 94 cm. Figure 10 shows the sensor positions on the ceiling.

4.2.2. Experimental Results of Sleep Posture Learning

The experimental results of the proposed sleep posture learning method are illustrated in Figure 15. Three subjects participated in this experiment. The human postures were evaluated using ultrasonic sensorsIn this regard, the proposed method learns the distance between sensors and determines the subject's posture without using any database sets. In processing the database sets, huge amounts of computing and processing power are consumed at the inference stage. Figure 15a presents the supine posture of subject 1, which is displayed on the FPGA Zed Board LCD display. The resulting sleep posture details are transmitted through the IoT module Wi-Fi ESP8266 to the monitoring or assisting team for their next course of assistance. Similarly, other postures of subject 1 were learned and registered as a reference for future usage, as in Figure 15c,d. The supine and right yearner postures of subject 2 are demonstrated in Figure 15e,f. Subject 3 was positioned in the left yearner posture and is displayed in Figure 15g,h. Figure 15b,d,h represent the supine, left lateral, and left yearner postures on the FPGA Zed-Board LCD display. This display is useful as interim information when transmitting posture details to the assisting team. The experimental results are presented as follows: https://www.youtube.com/watch?v=6nRHrVYnXTQ accessed on 14 September 2024.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**) (**g**) (**h**)

Figure 15. (**a**–**h**) Demonstration of results of sleep posture learning of various subjects. (**a**) Subject_1 Supine Posture. (**b**) Subject_1 Supine Posture Display. (**c**) Subject_1 Left Lateral Posture. (**d**) Subject_1 Left_Lateral_Display. (**e**) Subject_2 Supine Posture. (**f**) Subject_2 Right Yearner. (**g**) Subject_3 Left Yearner. (**h**) Subject_3 Left Yearner display.

4.2.3. Experimental Results of Adaptive Sleep Posture Analysis

After learning the sleep posture at the inference stage, the proposed approach provides a better solution for the estimation of posture analysis under both generic and adaptive event conditions. Figure 16a–f demonstrate the adaptive sleep postures of the initial subject in the experiment. In Figure 16a, the subject exhibits the left yearner (LY) posture. The subject changed to the right yearner (RY) pose, as shown in Figure 16c, and the supine posture (SP), as shown in Figure 16b. The same adaptive sleep posture taken during the interval of around 9.64 s from the LY end time to the RY start time was presented on the FPGA LCD display as TD (duration).

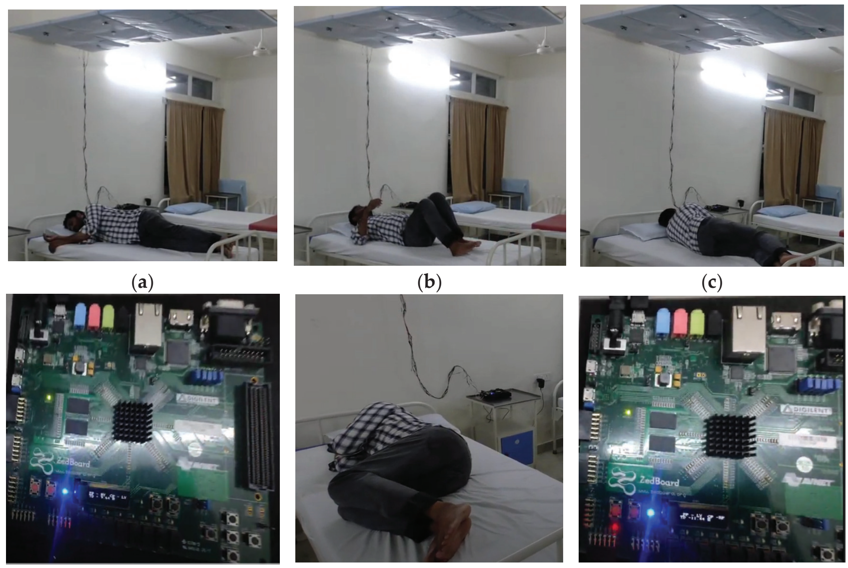


Figure 16. (**a**–**f**) Demonstration of results of adaptive sleep posture analysis.

The same subject switched his adaptive posture from right yearner (RY), as demonstrated in Figure 16c, to the left fetal (LF) position, as shown in Figure 16e, in the supine posture. The duration was recorded as 11.44 s. Figure 16d shows the past FPGA Zed-Board LCD display posture as right yearner (RY) and the current posture as left yearner (LY), along with the time duration. Similarly, 16f represents the adaptive posture of the left yearner to the right foetus. The experimental results are as follows (https://www.youtube.com/watch?v=Z8UvHXnd6lY accessed on 14 September 2024).

Table 4 presents a comparison with other research methods of sleep posture analysis. Most sleep posture detection methods have been developed for use with a bedsheet/mat. The authors of [17,25] used more sensors for accuracy, as any misleading data from sensors could impact the sleep posture analysis. However, this affected the power consumption. R. Tapwal et al. [26] utilized two costly and limited flex force sensors in a method that detected up to four postures only and proportionally consumed more power, at 17.5 watts. Hu, D et al. [27] utilized 32 piezoelectric ceramic sensors for analysis, achieving better results for nuanced pressure disturbances. The proposed methods perform sleep posture analysis in generic and adaptive scenarios, can be carried out using an FPGA-based accelerator with a low power consumption of around 1.2 watts, and can be operated using computing modules with a 100 MHz clock frequency; meanwhile, other comparison methods have a higher power consumption and require CPU resources. Data acquisition was performed using the Arduino Nano or Uno operated at 16 MHz. In the proposed method, a single reconfigurable device provided better results (98.4%) for both data acquisition and analysis.

Table 4. Comparison of sleep posture analysis with relevant research methods.

Reference Paper	Sensory Approach		Algorithm	Hardware	Number of Postures	Pros	Accuracy	Cons
	Method	Fusion						
Q. Hu et al., 2021 [17]	1024 pressure sensors	Yes	HOG, SVM, and CNN	Arduino Nano and CPU	6	<400 ms, sampling, and processing,	86.94% to 91.24%	Contact approach
Mater et al., 2020 [25]	1728 FSR sensors	Yes	HOG + LBP, FFANN	CPU	4	Health monitoring	97%	Increased usage of sensors
R. Tapwal et al., 2023 [26]	Two flex force sensors	Yes	K-means	Arduino Uno and CPU	4	Health monitoring	~99.3%	Consumes 17.5 W, contact approach
Hu, D et al., 2024 [27]	32 piezoelectric sensors	Yes	S^3CNN	N/A	4	Effectively detects nuanced pressure disturbances	93.0%	N/A
Proposed	6 ultrasonic sensors	Yes	HBC, heuristic learning	FPGA	8	Parallel computing, <370 ns, sampling, and computation.	98.4%	PR flow would be preferred in future usage

5. Conclusions

Sleep posture analysis has attracted considerable attention as a means of monitoring patients/children and the elderly. The proposed approach is the first of its kind to provide a solution with hardware schemes. Hardware schemes were adopted, alongside machine-learning-based heuristic methods, in the processing of sleep posture analysis at the learning, classification, and evaluation stages with processing elements (PEs). Sound-based data acquisition was successful in concurrently capturing and fusing data at a rate of 25 µs. The proposed method provides a better solution at the inference stage by using hardware schemes with adaptive subject sleep posture recognition and analysis with standard forms. This avoids excessive memory use at the learning and evaluation stages. Each subject and each posture-learning method was validated in 30 iterations, and the latency of the proposed hardware was around 370 ns. The results of the experiment showed 98.4% accuracy and a 1.6% error rate. The resource consumption of the optimized hardware schemes was 51% for the BRAM, 46% for the LUTs, and 44% for the DSP slices. Overall, 1.2 watts of power was consumed for computation. It is hoped that the device will be optimized in the future via partial reconfiguration methods and multi-subject sleep posture detection for hospital patients and senior citizens.

Author Contributions: Conceptualization, S.K.G., M.C.C., S.-K.L., G.D.V., N.J. and S.D.; methodology, M.S., S.K.G., M.C.C., S.-K.L., M.B. and S.D.; validation, M.S., M.C.C., G.D.V., M.B., D.H.K. and N.J.; writing—original draft preparation, M.S., M.C.C., S.K.G., D.H.K. and S.-K.L.; writing—review and editing, M.C.C., S.-K.L., S.K.G., G.D.V., N.J. and S.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is available within the article.

Acknowledgments: The EDA Tool and FPGA hardware were supported by the B. V. Raju Institute of Technology, Medak (Dist.), Narsapur.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Senaratna, C.V.; Perret, J.L.; Lodge, C.J.; Lowe, A.J.; Campbell, B.E.; Matheson, M.C.; Hamilton, G.S.; Dharmage, S.C. Prevalence of obstructive sleep apnea in the general population: A systematic review. *Sleep Med. Rev.* **2017**, *34*, 70–81. [CrossRef] [PubMed]
2. De Koninck, J.; Lorrain, D.; Gagnon, P. Sleep positions and position shifts in five age groups: An ontogenetic picture. *Sleep* **1992**, *15*, 143–149. [CrossRef] [PubMed]
3. Consensus Conference Panel; Watson, N.F.; Badr, M.S.; Belenky, G.; Bliwise, D.L.; Buxton, O.M.; Buysse, D.; Dinges, D.F.; Gangwisch, J.; Grandner, M.A.; et al. Joint consensus statement of the American Academy of Sleep Medicine and Sleep Research Society on the recommended amount of sleep for a healthy adult: Methodology and discussion. *J. Clin. Sleep Med.* **2015**, *11*, 931–952. [CrossRef]
4. Fallmann, S.; Chen, L. Computational Sleep Behavior Analysis: A Survey. *IEEE Access* **2019**, *7*, 142421–142440. [CrossRef]
5. Cheng, J.; Mu, Y.; Liu, C.; Yang, W.; Liu, W.; Wang, H.; Wu, J.; Hou, F.; Hao, D.; Cheng, L.; et al. A Fully Integrated Flexible Electronic System with Highly Sensitive MWCNTs Piezoresistive Array Sensors for Pressure Monitoring. *IEEE Sens. J.* **2022**, *22*, 18143–18150. [CrossRef]
6. Kau, L.-J.; Wang, M.-Y.; Zhou, H. Pressure-Sensor-Based Sleep Status and Quality Evaluation System. *IEEE Sens. J.* **2023**, *23*, 9739–9754. [CrossRef]
7. Zhang, Y.; Xiao, A.; Zheng, T.; Xiao, H.; Huang, R. The Relationship between Sleeping Position and Sleep Quality: A Flexible Sensor-Based Study. *Sensors* **2022**, *22*, 6220. [CrossRef]
8. Falch, L.; de Silva, C.W. Incorporating the Qualitative Variable Comfort into the Design of a Wearable Body Sensor System. *IEEE/ASME Trans. Mechatron.* **2021**, *26*, 124–133. [CrossRef]
9. Fallmann, S.; van Veen, R.; Chen, L.; Walker, D.; Chen, F.; Pan, C. Wearable accelerometer based extended sleep position recognition. In Proceedings of the 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), Dalian, China, 12–15 October 2017; pp. 1–6.
10. Renevey, P.; Delgado-Gonzalo, R.; Lemkaddem, A.; Proença, M.; Lemay, M.; Solà, J.; Tarniceriu, A.; Bertschi, M. Optical wristworn device for sleep monitoring. In *MBEC & NBC*; Springer: Singapore, 2017; pp. 615–618.
11. Chung, K.; Song, K.; Shin, K.; Sohn, J.; Cho, S.H.; Chang, J.-H. Noncontact sleep study by multi-modal sensor fusion. *Sensors* **2017**, *17*, 1685. [CrossRef]
12. Ong, A.A.; Gillespie, M.B. Overview of smartphone applications for sleep analysis. *World J. Otorhinolaryngol.-Head Neck Surg.* **2016**, *2*, 45–49. [CrossRef]
13. Saddaoui, R.; Gana, M.; Hamiche, H.; Laghrouche, M. Wireless Tag Sensor Network for Apnea Detection and Posture Recognition Using LSTM. *IEEE Embed. Syst. Lett.* **2024**. *Early Access*. [CrossRef]
14. Singh, S.; Anisi, M.H.; Jindal, A.; Jarchi, D. Smart Multimodal In-Bed Pose Estimation Framework Incorporating Generative Adversarial Neural Network. *IEEE J. Biomed. Health Inform.* **2024**, *28*, 3379–3388. [CrossRef] [PubMed]
15. Lin, B.-S.; Peng, C.-W.; Lee, I.-J.; Hsu, H.-K.; Lin, B.-S. System Based on Artificial Intelligence Edge Computing for Detecting Bedside Falls and Sleep Posture. *IEEE J. Biomed. Health Inform.* **2023**, *27*, 3549–3558. [CrossRef] [PubMed]
16. Liu, X.; Jiang, W.; Chen, S.; Xie, X.; Liu, H.; Cai, Q.; Tong, X.; Shi, T.; Qu, W. PosMonitor: Fine-Grained Sleep Posture Recognition With mmWave Radar. *IEEE Internet Things J.* **2024**, *11*, 11175–11189. [CrossRef]
17. Hu, Q.; Tang, X.; Tang, W. A Real-Time Patient-Specific Sleeping Posture Recognition System Using Pressure Sensitive Conductive Sheet and Transfer Learning. *IEEE Sens. J.* **2021**, *21*, 6869–6879. [CrossRef]
18. Liu, P.; Nguang, S.-K.; Partridge, A. Occupancy Inference Using Pyroelectric Infrared Sensors Through Hidden Markov Models. *IEEE Sens. J.* **2016**, *16*, 1062–1068. [CrossRef]
19. Gochoo, M.; Tan, T.-H.; Liu, S.-H.; Jean, F.-R.; Alnajjar, F.S.; Huang, S.-C. Unobtrusive Activity Recognition of Elderly People Living Alone Using Anonymous Binary Sensors and DCNN. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 693–702. [CrossRef]
20. Thathsara, M.; Lam, S.-K.; Kawshan, D.; Piyasena, D. Hardware Accelerator for Feature Matching with Binary Search Tree. In Proceedings of the 2024 IEEE International Symposium on Circuits and Systems (ISCAS), Singapore, 19–22 May 2024; pp. 1–5.
21. Schlegel, D.; Grisetti, G. HBST: A hamming distance embedding binary search tree for feature-based visual place recognition. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3741–3748. [CrossRef]
22. Nurvitadhi, E.; Sheffield, D.; Sim, J.; Mishra, A.; Venkatesh, G.; Marr, D. Accelerating Binarized Neural Networks: Comparison of FPGA, CPU, GPU, and ASIC. In Proceedings of the 2016 International Conference on Field-Programmable Technology (FPT), Xi'an, China, 7–9 December 2016; pp. 77–84.
23. Basha, M.; Kumar, M.S.; Chinnaiah, M.C.; Lam, S.-K.; Srikanthan, T.; Narambhatla, J.; Dodde, H.K.; Dubey, S. Hardware Schemes for Smarter Indoor Robotics to Prevent the Backing Crash Framework Using Field Programmable Gate Array-Based Multi-Robots. *Sensors* **2024**, *24*, 1724. [CrossRef]
24. Karumuri, S.R.; Lam, S.K.; Narambhatlu, J.; Dubey, S. Hardware-Efficient Scheme for Trailer Robot Parking by Truck Robot in an Indoor Environment with Rendezvous. *Sensors* **2023**, *23*, 5097. [CrossRef]
25. Matar, G.; Lina, J.-M.; Kaddoum, G. Artificial neural network for in-bed posture classification using bed-sheet pressure sensors. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 101–110. [CrossRef] [PubMed]

26. Tapwal, R.; Misra, S.; Deb, P.K. i-Sheet: A Low-Cost Bedsheet Sensor for Remote Diagnosis of Isolated Individuals. *IEEE Sens. J.* **2023**, *23*, 906–913. [CrossRef] [PubMed]
27. Hu, D.; Gao, W.; Ang, K.K.; Hu, M.; Chuai, G.; Huang, R. Smart Sleep Monitoring: Sparse Sensor-Based Spatiotemporal CNN for Sleep Posture Detection. *Sensors* **2024**, *24*, 4833. [CrossRef] [PubMed]

sensors

Article

Quantifying Hand Motion Complexity in Simulated Sailing Using Inertial Sensors

Gurdeep Sarai [1], Prem Prakash Jayaraman [2], Nilmini Wickramasinghe [3] and Oren Tirosh [1,4,*]

1 School of Health Sciences, Swinburne University of Technology, Hawthorn, VIC 3122, Australia
2 School of Science, Computing and Engineering Technologies, Swinburne University of Technology, Hawthorn, VIC 3122, Australia
3 School of Computing, Engineering and Mathematical Sciences, La Trobe University, Bundoora, VIC 3086, Australia
4 School of Health and Biomedical Sciences, Royal Melbourne Institute of Technology, Bundoora, VIC 3082, Australia
* Correspondence: oren.tirosh@rmit.edu.au

Abstract: The control of hand movement during sailing is important for performance. To quantify the amount of regularity and the unpredictability of hand fluctuations during the task, the mathematical algorithm Approximate Entropy (ApEn) of the hand acceleration can be used. Approximate Entropy is a mathematical algorithm that depends on the combination of two input parameters including (1) the length of the sequences to be compared (m), and (2) the tolerance threshold for accepting similar patterns between two segments (r). The aim of this study is to identify the proper combinations of 'm' and 'r' parameter values for ApEn measurement in the hand movement acceleration data during sailing. Inertial Measurement Units (IMUs) recorded acceleration data for both the mainsail (non-dominant) and tiller (dominant) hands across the X-, Y-, and Z-axes, as well as vector magnitude. ApEn values were computed for 24 parameter combinations, with 'm' ranging from 2 to 5 and 'r' from 0.10 to 0.50. The analysis revealed significant differences in acceleration ApEn regularity between the two hands, particularly along the Z-axis, where the mainsail hand exhibited higher entropy values (p = 0.000673), indicating greater acceleration complexity and unpredictability. In contrast, the tiller hand displayed more stable and predictable acceleration patterns, with lower ApEn values. ANOVA results confirmed that parameter 'm' had a significant effect on acceleration complexity for both hands, highlighting differing motor control demands between the mainsail and tiller hands. These findings demonstrate the utility of IMU sensors and ApEn in detecting nuanced variations in acceleration dynamics during sailing tasks. This research contributes to the understanding of hand-specific acceleration patterns in sailing and provides a foundation for further studies on adaptive sailing techniques and motor control strategies for both novice and expert sailors.

Keywords: approximate entropy (ApEn); inertial measurement units (IMUs); sailing simulation; motion time-series analysis; handedness

Citation: Sarai, G.; Jayaraman, P.P.; Wickramasinghe, N.; Tirosh, O. Quantifying Hand Motion Complexity in Simulated Sailing Using Inertial Sensors. *Sensors* **2024**, *24*, 6728. https://doi.org/10.3390/s24206728

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 2 September 2024
Revised: 15 October 2024
Accepted: 16 October 2024
Published: 19 October 2024

1. Introduction

The subtle dance of our fingers, the precise movements of our hands, fine motor control is woven into nearly every aspect of human life. It involves the coordination of small muscles in the eyes, hands, and fingers, leading to dynamic movements such as reaching and grasping [1]. This intricate coordination extends beyond everyday tasks, playing a crucial role in various physical activities. In the context of sailing, achieving optimal wind utilization necessitates fine motor control for precise adjustments to the sails [2]. These adjustments can be interpreted as "reaching" and "side-to-side" tasks, which are critical for the performance and safety of sailors. Investigating these movements and handedness provides valuable insights into both healthy individuals and those with various medical conditions [3].

Handedness refers to the tendency to use one hand over the other and is linked to brain lateralization, with each hemisphere controlling movement in the opposite hand [4]. It includes right-handedness, left-handedness, and ambidexterity. Handedness consists of two components: hand preference (using the preferred hand for tasks) and hand performance (differences in abilities between hands) [5,6]. Most individuals develop a hand preference from a young age, influenced by genetic and environmental factors [7]. Generally, the dominant hand is more adept at performing skilled tasks that require precision and fine motor control, such as steering or aiming [8], whereas the non-dominant hand is frequently tasked with supporting roles that involve broader, less refined motions, such as holding or stabilizing. However, unskilled tasks like reaching may show no significant difference between both hands [9,10]. This distinction between hand functions can be impactful when assigning roles in complex activities like sailing, where the precision and sensitivity required for steering (tiller control) are better suited to the dominant hand, while the less intricate but forceful maneuvers needed to handle the mainsail are effectively managed by the non-dominant hand. However, the relationship between hand skill and preference remains complex and not fully understood, potentially due to variations dependent on experience and environmental demands.

Research on motor skill lateralization has focused on how specialization contributes to differences in hand preference and performance. Hemispheric asymmetry, the division of control between brain hemispheres, has also been explored. For instance, Nelson et al. [11] investigated handedness and limb control in adults, finding mixed support for the dynamic dominance hypothesis, with right-handers generally performing better. Mutha et al.'s [12] study challenges conventional handedness views, suggesting specialized mechanisms in each arm–hemisphere system, with the dominant hemisphere aiding predictive control and the non-dominant one stabilizing the arm at a goal position. Boulinguez-Ambroise et al. [13] highlighted that although hemispheric dominance is an important factor, it does not fully explain the variations observed in performance. This suggests that additional cognitive and motor elements also contribute to the underlying mechanisms. Overall, the interplay between hemispheric asymmetry and motor skill specialization is complex and multidimensional, necessitating further research to comprehensively understand its implications for hand preference and performance.

Hand function assessments frequently rely on questionnaires, which can introduce subjectivity and bias. As noted by Mcsp and Dipcot [14], and further supported by Peters [15], self-reported measures often fail to accurately capture an individual's actual hand capabilities due to personal perceptions and emotional states [14,15]. This subjectivity can lead to inconsistent results, particularly in clinical settings where objective measures are crucial. Bear-Lehman and Abreu [16] also highlight that questionnaires may overlook essential aspects of hand function, such as dexterity and strength [16]. This highlights the importance of quantitative data for studying movement patterns between hands as well as choosing the appropriate technology and measures for accurately capturing fine motor control and dynamic hand movements [17].

Entropy, a measure of system randomness or uncertainty, is one such measure [18]. Numerous studies have used Inertial Measuring Units (IMUs) to measure reaching, applying kinematics and kinetics techniques. However, analyzing three-axis motion time-series data for hand dominance differences in sailing has been overlooked, likely due to the complexity of such data.

This study seeks to address the gap in research by examining the control of the hand movement during multi-directional sailing simulation. Movement control of the hand can be quantified using the 'regularity statistic' Approximate Entropy (ApEn) that quantifies the unpredictability of fluctuations in hand acceleration time series. ApEn reflects the likelihood that 'similar' patterns of observations will not be followed by additional 'similar' observations. Acceleration time series containing many repetitive patterns have a relatively small ApEn; a less predictable and more complex process has a higher ApEn value. The algorithm for computing ApEn depends on the 'm' and 'r' input parameters. The parameter 'm' specifies the pattern length, and 'r' defines the criterion of similarities. The aim of this study is to identify the proper combinations of 'm' and 'r' parameter values for ApEn measurement in the hand movement acceleration data during sailing. The findings will provide valuable insights to inform future investigations on the appropriate selection of parameters for ApEn analyses in sailing.

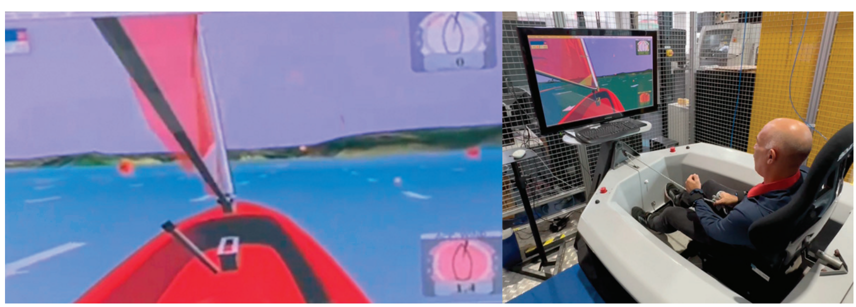
2. Method

2.1. Participants

The study involved 10 participants, aged 18 to 30 years, who volunteered after an initial screening process. Eligibility criteria included no prior experience in sailing, and the absence of current or chronic health conditions and physical disabilities or injuries. Individuals with clinically significant diseases or disorders were excluded. Participants underwent further screening using the Adult Pre-Exercise Screening System [19] tool to identify those susceptible to adverse exercise-related events and those showing signs of known illnesses. Hand dominance was determined using the Edinburgh Handedness Inventory [20], revealing that eight participants were right-handed and two were left-handed. Prior to the experiment, all participants received an information sheet detailing the research process and provided informed consent. Ethical approval was granted by the Swinburne University Human Research Ethics Committee (SUHREC—Ref: 20226403-12100).

2.2. Equipment

The study utilized a VSail-Trainer®, designed by the company Virtual Sailing Pty Ltd. (Virtual Sailing, Parkville, VIC, 3052, Australia). It comprises one boat hull (size length: 230 cm, breadth: 150 cm) based on the Hansa 303 model boat (see Figure 1). The sailor controls the simulator in a seated setup, controlling the virtual Hansa dinghy's course via a centrally located joystick and managing speed through the mainsheet. A 42-inch screen displayed the virtual course, while sensors tracked the heel angle, tiller angle, and mainsail tension. This real-time data enabled adjustments to the simulated sailing environment, including wind conditions, gusts, and course variations.



a **b**

Figure 1. (**a**) The virtual HUD display in the sailing simulator, also showing the triangular course. (**b**) A participant during a trial with the sailing simulator.

The experimental setup was designed to simulate the ergonomics and control mechanisms of seated dinghy sailing, specifically the Hansa dinghy. A centrally positioned joystick is a common adaptation in accessible sailing for individuals with physical disabilities or those new to sailing. This choice aimed to minimize physical strain and enhance participant engagement.

This study used 2 wireless IMUs, with tri-axial accelerometers (YEI 3-space sensor, Yost Labs, Portsmouth, OH 45662, USA) to capture continuous hand acceleration during sailing. These lightweight sensors (35 mm $\times$ 60 mm $\times$ 15 mm, 28 g) were secured to participants' hands using Velcro straps (see Figure 2). Raw acceleration was sampled at

150 Hz (±16 g range) and captured using in-house Python software (version 2.7). The IMU has ±1° orientation accuracy for dynamic conditions, with <0.08° resolution and 0.085° repeatability across all axes.

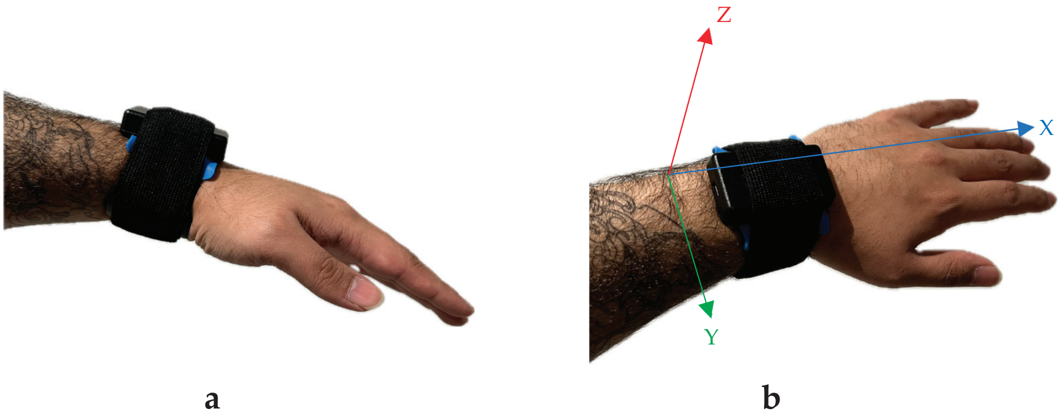


Figure 2. The placement of the IMU sensors on the left and right wrist. IMU reference: the X-axis is pointing forwards, the Y-axis is pointing laterally, and the Z-axis is for the depth. (**a**) Lateral View. (**b**) Point of View.

2.3. Multi-Directional Sailing Task

Participants performed the multi-directional sailing simulation using both dominant and non-dominant hands. The course was based on the Sydney 2000 Olympic Games triangle course in Sydney Harbor, designed to challenge participants with upwind and downwind segments. The sailing simulation course commenced with a standardized 30 s start protocol, characterized by northerly winds of 12 knots blowing towards the south. While the simulator dynamically adjusted to user input, ensuring a realistic sailing experience, each participant completed the same pre-defined course involving two tacks and one gybe. To further standardize the sailing task, a "ghost" boat, representing the ideal course trajectory, was incorporated into the visual display. This provided participants with a real-time visual guide for the pre-defined course, ensuring consistent navigational cues throughout the experiment.

This consistency in course maneuvers ensured that all participants encountered identical navigational challenges. Variations in individual performance stemmed from differences in the tiller steering, mainsail trim (specifically twin-tail adjustment), and overall reaction to the simulated conditions, rather than variations in the course itself.

2.4. Procedure

Participants were equipped and strapped with an IMU on both wrists to measure accelerations during the sailing task (see Figure 2). Prior to testing, participants had a familiarization session to familiarize themselves with the V-Sail simulator. During the familiarization session, participants received instruction on sailing techniques, including the use of the Heads-Up Display (HUD), which provided real-time data on the wind direction and strength, twin tails' alignment, boat angle, and course details. The participants were also instructed on the placement of hands, where the dominant hand was typically assigned to the tiller, as steering requires precise, fine motor adjustments. The non-dominant hand was tasked with adjusting the mainsail, a function that, while critical, does not demand the same level of precision as steering in the beginning stages. This division of labor reflects common beginner sailing practices, where the dominant hand's dexterity is leveraged for steering, and the non-dominant hand supports sail handling, helping the participant during the initial learning process.

During the familiarization session, participants were taught to steer the boat using the tiller, navigate to specific objects, and maneuver around obstacles. They then learned to adjust the mainsail to optimize the sail angle for speed and maneuverability. After mastering basic sailing techniques, participants practiced specific maneuvers such as tacking and gybing, crucial for sailing against the wind or overcoming headwinds. The final stage involved managing the heeling angle using pneumatic rams to reduce water resistance and improve speed and control. The instructional procedures during the familiarization session were standardized across all participants to ensure that the duration of practice and the feedback provided were consistent across participants to eliminate any confounding effects.

Following the familiarization session, participants performed the testing session where hand acceleration data were captured. During the testing session, participants were required to complete 5 laps of the triangular course, navigating outside the buoys, and cross the finishing line as quickly as possible. The various performance measures were collected using the wireless IMUs.

2.5. Entropy Measures of Acceleration Time-Series Data

The computation of ApEn, a pivotal metric in assessing motor control complexities, hinges on several intricate parameters crucial for a meaningful analysis [21]. Among these, 'm' and 'r' are the primary determinants. The parameter 'm' signifies the number of sequences being compared for similarity, i.e., m = the length of embedding vectors, while 'r' sets the threshold for considering sequences as similar. Additionally, secondary determinants such as 'N' (number of data points) ensure a reliable estimate, while the time delay establishes the temporal gap between compared points. The window size influences the analysis scope by determining the portion of data analyzed at any given time. Data normalization is employed to remove scale-induced bias. These parameters collectively ensure the reliability and relevance of entropy calculations, requiring adjustments to align with specific research objectives and the nature of the data.

Our study primarily concentrates on the primary determinants 'm' and 'r', as these are directly modifiable and crucial for computing ApEn. These factors are essential for balancing the identification of true complexity against the risk of misinterpretation due to noise or inadequate data capture [21]. While other parameters like the N (total number of data points), time delay, window size, and data normalization play supportive roles, they either pertain to the broader study design, pre-processing requirements, or specific considerations of the ApEn algorithm.

In the context of our research using a sailing simulator, identifying and fine-tuning 'm' and 'r' are paramount before addressing secondary determinants. A well-chosen 'm' ensures that the complexity analysis captures relevant hand patterns, while an appropriately set 'r' distinguishes true behavioral variability from random noise. This allows for a better understanding of motor control stability or irregularity in individuals during the sailing simulator task.

For reliable entropy results, 'm' is typically selected as 2, with occasional use of 3 [22–24]. However, we decided to extend this to 5 to explore the potential influence of a higher embedding dimension on entropy measures in our sailing simulator task. These 'm' values enable the analysis of sequential complexity from basic (m = 2, three consecutive data points) to advanced (m = 5, patterns of six consecutive data points), ensuring the capture of both simple repetitive patterns and more complex structures in the data.

Typically, values of 'r' like 0.1 or 0.2 are common [25,26]. However, due to the unique application of the sailing simulator, we explored higher values up to 0.4. The chosen 'r' values (0.1 to 0.4) define the tolerance spectrum for matching data segments. A lower 'r' (0.10) enhances sensitivity to fine details, detecting subtle variations, while a higher 'r' (0.40) smooths over minor differences, aiding in recognizing broader patterns and reducing sensitivity to noise.

2.6. Statistical Procedures

The data processing procedures in this study followed well-established protocols to ensure an accurate analysis of the Inertial Measurement Unit data collected during the sailing simulation tasks.

1. The raw acceleration data from the IMUs were processed to remove noise and baseline drift [27].
2. A zero-phase Butterworth high-pass filter with a cut-off frequency of 0.3 Hz was applied to the acceleration data to eliminate low-frequency noise, ensuring that only the relevant motion data were retained for a further analysis.
3. Following the high-pass filtering, the data underwent smoothing using a third-order Savitzky–Golay filter with a frame size of 41 points. This filter was chosen to preserve the features of the data while reducing high-frequency noise, making it suitable for the entropy analysis.
4. Approximate Entropy (ApEn) values were then calculated, employing the method described by [28]. ApEn was computed using varying combinations of the embedding dimension 'm' (values 2 to 5) and the similarity threshold 'r' (values 0.1 to 0.5), applied across all three axes (X, Y, Z), and the vector magnitude (Vm) of the hand motion data. The approx_entropy function from the R 'pracma' library was used to perform these calculations.
5. A one-way repeated measures ANOVA was conducted to statistically evaluate the significance of differences in Approximate Entropy values between the tiller hand and the mainsail hand. This analysis was performed separately for each of the four variables: the X-axis, Y-axis, Z-axis, and vector magnitude. The analysis aimed to identify any significant differences in the complexity of hand motion across these axes.
6. To further investigate the significant effects observed in the one-way repeated measures ANOVA, a post hoc Tukey's Honest Significant Difference test was performed. This allowed for targeted, pairwise comparisons between the conditions while upholding the appropriate statistical standards after the initial ANOVA analysis.

3. Results

To compute Approximate Entropy (ApEn) values from the motion time-series data, we experimented with various combinations of parameter values, specifically 'm' values ranging from 2 to 5, and 'r' values between 0.10 and 0.50. These computations were applied to the X-axis, Y-axis, Z-axis, and vector magnitude of both dominant and non-dominant hand motion data, resulting in a total of 24 parameter combinations for each ApEn analysis.

The mean and standard deviation (SD) of ApEn values for tiller (dominant) and mainsail (non-dominant) hand acceleration patterns are presented in Table A1. The analysis of variance (ANOVA) was conducted to assess the impact of both tiller and mainsail hands, along with factors M and R, on ApEn values across four variables: X, Y, Z, and vector magnitude (Vm). The results are detailed below.

For variable X, the ANOVA revealed no statistically significant difference between the tiller and mainsail hands ($p = 0.906$). This high p-value suggests that the mean ApEn values for variable X do not differ significantly between the two. However, the factor M exhibited a highly significant effect on X, with a p-value of less than 0.001 ($p = 4.03 \times 10^{-5}$), indicating that different levels of M contribute substantially to the variance in X.

While the analysis of variable Y yielded a p-value of 0.0851, suggesting a potential difference between tiller and mainsail hand acceleration patterns, this difference did not reach statistical significance at the conventional $\alpha = 0.05$ level, indicating that any difference in mean ApEn values between both hands may be minor. In contrast, the factor M had a highly significant impact on Y ($p = 7.41 \times 10^{-5}$), similar to its effect on X.

Variable Z showed a statistically significant difference between the tiller and mainsail hands ($p = 0.000673$), highlighting a strong difference in ApEn values between the two hands. This finding suggests that the complexity of hand acceleration patterns, as measured by ApEn, varies significantly between the tiller and mainsail hands on the Z-axis. Additionally, the factor M had a highly significant effect on Z, with an extremely low *p*-value ($p = 5.36 \times 10^{-7}$), further emphasizing its influence. The small residuals observed indicate that the model explains most of the variance in Z.

For variable Vm, the ANOVA results indicated no statistically significant difference between the tiller and mainsail hands ($p = 0.145$). This suggests that the mean ApEn values for vector magnitude do not differ significantly between the two groups. Nevertheless, similar to the other variables, the factor M demonstrated a highly significant effect on Vm ($p = 2.07 \times 10^{-5}$), suggesting that M consistently influences the variance across different axes.

Given the significant ANOVA result for variable Z, a post hoc Tukey's Honest Significant Difference (HSD) test was conducted. The Tukey's HSD test confirmed the ANOVA results, revealing that the ApEn values for the mainsail (non-dominant) hand were significantly higher than those for the tiller (dominant) hand on the Z-axis ($p = 0.000673$). These findings suggest that the mainsail hand exhibits greater variability and less predictability in its acceleration patterns during the sailing simulation task, in contrast to the dominant tiller hand.

The influence of the 'r' parameter on ApEn values was examined across all measured variables. As expected, elevating the 'r' parameter, which widens the tolerance for resemblance between data sequences, typically led to lower Approximate Entropy values. This was observed consistently across all parameters such as the X-axis tiller hand mean (decreasing from 0.272 at R0.1 to 0.028 at R0.4) and Y-axis mainsail hand mean (decreasing from 0.282 at R0.1 to 0.020 at R0.4).

Table A1—ApEn values for the X-axis, Y-axis, Z-axis, and vector magnitude as a function of the 20-point input parameter combinations m and r, for dominant and non-dominant hands, in terms of the mean and standard deviation.

4. Discussion

This study explores entropy variations between tiller (dominant) and mainsail (non-dominant) hand acceleration patterns within a multi-directional sailing simulation task. Using ApEn, a well-established tool for assessing complexity and predictability in time-series data, our research reveals subtle yet significant differences in the regularity and complexity of acceleration profiles in this simulated environment.

Our study introduces a task with three-dimensional freedom, facilitating hand acceleration analyses across the Y-, X-, and Z-axes. This innovative sailing simulation necessitated a more thorough investigation of acceleration, particularly as the Z-axis exhibited notable differences in the complexity of acceleration between the tiller and mainsail hands. While the present sailing simulation task shared some commonalities with traditional reaching paradigms, it incorporated additional elements to mitigate variability and maintain consistency with established reaching protocols. Specifically, the simulation featured a "ghost" boat that provided a target course, encouraging participants to minimize deviations from the pre-defined path. This is similar to reaching tasks, where hand movements typically exhibit uniformity and follow pre-defined trajectories with minimal variability. In contrast, the dynamic nature of sailing, which requires continuous adjustments due to factors such as wind and boat behavior, does result in less predictable and more varied hand movements. As a result, previous research indicating parameters used in reaching tasks may not apply to the dynamic nature of sailing. Our research highlights the need for sailing-specific parameters to capture these nuanced hand accelerations, which differ based on sailing complexity.

Our analysis of the sailing task revealed distinct entropy variations, with notably higher ApEn values in the mainsail (non-dominant) hand. These elevated values indicate greater irregularity and unpredictability in the mainsail hand's acceleration patterns, likely due to the complex demands of sailing. The tiller (dominant) hand, receiving more attention and training in daily activities, exhibited enhanced motor control and precision, reflected in its lower ApEn values across the X-, Y-, and Z-axes, as well as the vector magnitude. Whilst there are no previous studies on entropy for both hands in sailing, research has indicated that inexperienced sailors exhibit a limited ability to adapt to changing conditions, which may lead to less coordinated acceleration patterns between their hands during maneuvers [25]. This is particularly relevant as inexperienced sailors often struggle with the complexities of sail handling, resulting in more pronounced and exaggerated acceleration patterns of the mainsail hand to manage the sail effectively [26]. While expert sailors exhibit more refined and coordinated acceleration profiles, novices are likely to overcompensate with their mainsail hand, leading to a higher incidence of upper-limb overuse injuries [27]. Therefore, our results align with previous findings suggesting that inexperienced sailors primarily use their mainsail hand more than their tiller hand during sailing.

The parameter ‘r’ in the Approximate Entropy algorithm represents the tolerance or similarity threshold used to determine whether two data sequences are considered sufficiently alike to be classified as matching or resembling one another. The findings indicated that increasing the ‘r’ parameter typically resulted in a more smooth and less complex representation of the data. However, the rate of ApEn decreasing varied depending on the specific parameter and axis, indicating differential sensitivity to changes in ‘r’. For instance, while the X-axis tiller hand mean exhibited a substantial decrease, the Vm tiller hand mean showed a less pronounced reduction. These findings highlight the importance of considering parameter-specific responses to ‘r’ when interpreting ApEn values and underscore that larger similarity thresholds can mask subtle variations in data complexity. As the sailing task itself is very complex but also very sensitive, maintaining a low ‘r’ value is vital to ensure the identification of nuanced differences in hand acceleration patterns. However, if this task was less sensitive and required larger, slower acceleration profiles, a larger ‘r’ value may be more appropriate to smooth over the natural variability and identify the broader patterns.

The parameter ‘m’ in the Approximate Entropy algorithm represents the length of the sequences being compared to assess similarity within the data. The analysis revealed that increasing the ‘m’ parameter generally led to lower ApEn values, reflecting a more stringent and less complex representation of the data. These findings also emphasize the importance of carefully selecting the ‘m’ parameter to capture the appropriate level of detail in the data. For tasks that require sensitivity to subtle variations in acceleration patterns, such as the sailing simulation task described, a lower ‘m’ value for the Approximate Entropy algorithm may be necessary. This is because the sailing task involves irregular acceleration, necessitating the ability to capture subtle details in the data. Conversely, for other tasks where the goal is to identify broader, more consistent acceleration profiles, a higher ‘m’ value may be more appropriate, as it can provide a less complex, smoother representation of the data.

The Approximate Entropy metric has demonstrated its effectiveness in analyzing motion time-series data, as evidenced by its ability to distinguish between the tiller and mainsail hand acceleration patterns, particularly on the Z-axis, as confirmed by the post hoc Tukey’s HSD test [28]. Additionally, ApEn was capable of capturing a wide range of hand acceleration profiles, from subtle, precise changes to more broad, sweeping shifts. Experimenting with ‘m’ values (2, 3, 4, and 5) and ‘r’ values ranging from 0.10 to 0.50 provided insights. The factors ‘m’ and ‘r’ consistently had a significant impact across all variables (X, Y, Z, Vm), highlighting their influence on the complexity of hand acceleration patterns. These findings offer valuable guidance for researchers aiming to optimize ‘m’ and ‘r’ parameter selections when using ApEn to gauge hand acceleration complexity in sailing. Our research suggests that low ‘m’ (2) and ‘r’ (0.1) parameter values are sufficient to ensure an adequate number of sequence vectors within the tolerance for estimating conditional probabilities in our sailing simulation task.

The present study lays the foundation and key results for future research investigating the use of IMUs in analyzing motion time-series data. Such analyses could benefit from the application of Approximate Entropy due to its capacity to detect complexities and irregularities in multi-directional sailing tasks. The parameters selected for ApEn calculation should be carefully considered and varied to explore the sensitivity of the method in detecting differences in hand acceleration profiles. Our findings indicate that in the context of sailing using a VSail simulator, selecting and adjusting ApEn parameters diligently are essential. Optimal parameter combinations, such as m = 2, 3, 4, and 5 with r = 0.15, 0.20, 0.30, 0.40, and 0.50, were identified as effective in detecting nuances in hand acceleration differences and should be specifically utilized when examining reaching data within this domain. By introducing a multi-directional reaching task using sailing, our research offers valuable opportunities for functional diagnostics and assessments for multiple populations. Future studies could also explore more flexible hand-task assignments or how more experienced sailors adapt to sailing operations without hand assignments.

5. Conclusions

In conclusion, this study highlights the importance of Approximate Entropy (ApEn) as a tool for analyzing the complexities of hand acceleration patterns in a sailing simulation task. The findings reveal significant differences in the regularity and predictability of acceleration patterns between the tiller (dominant) and mainsail (non-dominant) hands, with the mainsail hand exhibiting higher ApEn values, suggestive of greater variability and complexity. The findings emphasize the importance of meticulously selecting and adjusting the 'm' and 'r' parameters in an Approximate Entropy analysis to effectively discern the subtleties of motor control during these types of tasks. The study establishes a foundation for future research, particularly in the context of using IMUs to explore motion time-series data in multi-directional tasks. By applying these insights, further exploration into hand acceleration dynamics can enhance our understanding of motor control in both healthy and clinical populations.

Author Contributions: Conceptualization, O.T., G.S., P.P.J. and N.W.; methodology, O.T., G.S., P.P.J. and N.W.; formal analysis, O.T., G.S., P.P.J. and N.W.; investigation, G.S., P.P.J. and N.W.; resources, O.T., P.P.J. and N.W.; data curation, O.T., G.S., P.P.J. and N.W.; writing—original draft preparation, O.T. and G.S.; writing—review and editing, O.T., G.S., P.P.J. and N.W.; visualization, G.S. and O.T.; supervision, O.T., P.P.J. and N.W.; project administration, O.T., P.P.J. and N.W.; funding acquisition, O.T., P.P.J. and N.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from Swinburne University of Technology the SUPRA scheme, in partnership with Northern Health and Virtual Sailing.

Institutional Review Board Statement: The study was approved by Swinburne University of Technology Research Ethic committee HREC 20226403-12100.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: This endeavor would not have been possible without the support of the Virtual Sailing team Norman Saunders, Mark Habgood, and Jonathan Binns, along with the added support of Northern Health from Peter Brooks.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. The mean and standard deviation of ApEn values for the X-axis, Y-axis, Z-axis, and vector magnitude of motion time-series data, across a range of 'm' and 'r' parameter combinations, for both tiller (dominant) and mainsail (non-dominant) hand movements.

M	R	Approximate Entropy															
		X				Y				Z				Vm			
		Dominant		Non-Dominant		Dominant		Non-Dominant		Dominant		Non-Dominant		Dominant		Non-Dominant	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
2	0.1	0.272	0.052	0.283	0.068	0.261	0.028	0.283	0.054	0.204	0.022	0.212	0.028	0.428	0.055	0.453	0.086
	0.15	0.143	0.038	0.147	0.043	0.136	0.029	0.148	0.038	0.095	0.014	0.103	0.023	0.220	0.043	0.233	0.056
	0.2	0.083	0.023	0.091	0.033	0.075	0.021	0.090	0.026	0.054	0.009	0.059	0.016	0.126	0.028	0.143	0.040
	0.3	0.049	0.014	0.055	0.024	0.042	0.014	0.052	0.017	0.029	0.008	0.035	0.014	0.071	0.018	0.086	0.027
	0.4	0.029	0.012	0.037	0.018	0.027	0.010	0.034	0.011	0.017	0.007	0.019	0.008	0.044	0.014	0.055	0.018
3	0.1	0.247	0.054	0.248	0.060	0.245	0.039	0.254	0.048	0.187	0.019	0.197	0.034	0.396	0.063	0.407	0.081
	0.15	0.130	0.036	0.136	0.041	0.121	0.027	0.129	0.037	0.083	0.012	0.092	0.022	0.198	0.041	0.209	0.057
	0.2	0.073	0.022	0.082	0.033	0.066	0.020	0.078	0.024	0.046	0.008	0.051	0.015	0.109	0.027	0.126	0.040
	0.3	0.040	0.013	0.050	0.025	0.036	0.013	0.046	0.015	0.025	0.007	0.030	0.012	0.060	0.018	0.076	0.027
	0.4	0.025	0.010	0.035	0.016	0.023	0.009	0.030	0.010	0.015	0.006	0.017	0.007	0.038	0.013	0.050	0.018
4	0.1	0.227	0.049	0.223	0.053	0.228	0.035	0.233	0.043	0.175	0.018	0.183	0.034	0.367	0.057	0.371	0.073
	0.15	0.113	0.034	0.117	0.040	0.108	0.024	0.113	0.034	0.079	0.019	0.083	0.022	0.177	0.039	0.184	0.053
	0.2	0.064	0.021	0.072	0.029	0.059	0.018	0.069	0.022	0.040	0.007	0.045	0.014	0.097	0.025	0.111	0.036
	0.3	0.035	0.011	0.046	0.021	0.032	0.012	0.041	0.014	0.022	0.006	0.025	0.009	0.053	0.016	0.067	0.025
	0.4	0.021	0.009	0.032	0.015	0.020	0.008	0.027	0.010	0.014	0.006	0.015	0.006	0.033	0.012	0.045	0.017
5	0.1	0.209	0.046	0.199	0.053	0.209	0.038	0.207	0.052	0.160	0.029	0.165	0.042	0.337	0.062	0.332	0.082
	0.15	0.100	0.029	0.107	0.039	0.098	0.022	0.108	0.031	0.069	0.010	0.077	0.021	0.157	0.033	0.171	0.051
	0.2	0.058	0.019	0.063	0.027	0.054	0.016	0.062	0.021	0.037	0.006	0.041	0.013	0.088	0.023	0.099	0.033
	0.3	0.033	0.011	0.042	0.018	0.030	0.012	0.038	0.011	0.020	0.006	0.024	0.008	0.050	0.016	0.063	0.021
	0.4	0.019	0.008	0.029	0.014	0.018	0.008	0.025	0.009	0.012	0.005	0.014	0.006	0.029	0.011	0.041	0.015

References

1. Niechwiej-Szwedo, E.; Wu, S.; Nouredanesh, M.; Tung, J.; Christian, L.W. Development of eye-hand coordination in typically developing children and adolescents assessed using a reach-to-grasp sequencing task. *Hum. Mov. Sci.* **2021**, *80*, 102868. [CrossRef] [PubMed]
2. Altosole, M.; Piastra, F.; Canepa, E. Performance analysis of a motor-sailing propulsion system for control design purposes. *Ships Offshore Struct.* **2016**, *11*, 688–699. [CrossRef]
3. Pool, E.M.; Rehme, A.K.; Fink, G.R.; Eickhoff, S.B.; Grefkes, C. Handedness and effective connectivity of the motor system. *Neuroimage* **2014**, *99*, 451–460. [CrossRef] [PubMed]
4. Vaid, J.; Bellugi, U.; Poizner, H. Hand dominance for signing: Clues to brain lateralization of language. *Neuropsychologia* **1989**, *27*, 949–960. [CrossRef] [PubMed]
5. Mundorf, A.; Getzmann, S.; Gajewski, P.D.; Larra, M.F.; Wascher, E.; Ocklenburg, S. Stress exposure, hand preference, and hand skill: A deep phenotyping approach. *Laterality* **2023**, *28*, 209–237. [CrossRef]
6. Scharoun, S.M.; Bryden, P.J. Hand preference, performance abilities, and hand selection in children. *Front. Psychol.* **2014**, *5*, 82. [CrossRef]
7. Babik, I.; Lobo, M.A. Hand-Use Preferences for Reaching and Object Exploration in Children with Impaired Upper Extremity Functioning: The Role of Environmental Affordances. *Symmetry* **2023**, *15*, 2161. [CrossRef]
8. Coren, S. Sensorimotor performance as a function of eye dominance and handedness. *Percept. Mot. Ski.* **1999**, *88*, 424–426. [CrossRef]
9. Cordova Bulens, D.; Cluff, T.; Blondeau, L.; Moore, R.T.; Lefevre, P.; Crevecoeur, F. Different Control Strategies Drive Interlimb Differences in Performance and Adaptation during Reaching Movements in Novel Dynamics. *Eneuro* **2023**, *10*. [CrossRef]
10. Peters, M. Constraints in the performance of bimanual tasks and their expression in unskilled and skilled subjects. *Q. J. Exp. Psychol. Sect. A* **1985**, *37*, 171–196. [CrossRef]
11. Nelson, E.L.; Berthier, N.E.; Konidaris, G.D. Handedness and Reach-to-Place Kinematics in Adults: Left-Handers Are Not Reversed Right-Handers. *J. Mot. Behav.* **2018**, *50*, 381–391. [CrossRef] [PubMed]
12. Mutha, P.K.; Haaland, K.Y.; Sainburg, R.L. The effects of brain lateralization on motor control and adaptation. *J. Mot. Behav.* **2012**, *44*, 455–469. [CrossRef] [PubMed]
13. Boulinguez-Ambroise, G.; Aychet, J.; Pouydebat, E. Limb Preference in Animals: New Insights into the Evolution of Manual Laterality in Hominids. *Symmetry* **2022**, *14*, 96. [CrossRef]
14. Mcsp, I.C.B.; Dipcot, J.A. A Comparison of Dominant and Non-dominant Hand Function in both Right- and Left-Handed Individuals using the Southampton Hand Assessment Procedure (SHAP). *Br. J. Hand Ther.* **2003**, *8*, 4–10. [CrossRef]
15. Peters, M. Description and Validation of a Flexible and Broadly Usable Handedness Questionnaire. *Laterality* **1998**, *3*, 77–96. [CrossRef]
16. Bear-Lehman, J.; Abreu, B.C. Evaluating the hand: Issues in reliability and validity. *Phys. Ther.* **1989**, *69*, 1025–1033. [CrossRef]
17. Choo, M.; Yu Zheng, C.; Yea Dat, C.; Loo, J. *Evaluation of Hand Functions Using Sensor-Based Wearable Hand Motion Analysis Device*; IEEE: Piscataway, NJ, USA, 2023; pp. 26–31.
18. Delgado-Bonal, A.; Marshak, A. Approximate Entropy and Sample Entropy: A Comprehensive Tutorial. *Entropy* **2019**, *21*, 541. [CrossRef]
19. Perera, N.; Keyzer, P.; Dietrich, J.; Norton, K.; Sekendiz, B.; Jones, V.; Coyle, I.; Finch, C. Awareness and use of the adult pre-exercise screening system (apss) in the australian fitness industry. *Br. J. Sports Med.* **2014**, *48*, 651. [CrossRef]
20. Oldfield, R.C. *Edinburgh Handedness Inventory [Database record]*; APA PsycTests; American Psychological Association: Washington, DC, USA, 1971; Volume 9, pp. 97–113.
21. Reddy, C.; Reddy, M. *Sensitivity of Threshold Value 'r' in Approximate Entropy Based Brain-Computer Interface*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6.
22. Estrada, L.; Torres, A.; Sarlabous, L.; Jané, R. Influence of Parameter Selection in Fixed Sample Entropy of Surface Diaphragm Electromyography for Estimating Respiratory Activity. *Entropy* **2017**, *19*, 460. [CrossRef]
23. Restrepo, J.F.; Schlotthauer, G.; Torres, M.E. Maximum approximate entropy and r threshold: A new approach for regularity changes detection. *Phys. A Stat. Mech. Its Appl.* **2014**, *409*, 97–109. [CrossRef]
24. Yentes, J.M.; Hunt, N.; Schmid, K.K.; Kaipust, J.P.; McGrath, D.; Stergiou, N. The appropriate use of approximate entropy and sample entropy with short data sets. *Ann. Biomed. Eng.* **2013**, *41*, 349–365. [CrossRef] [PubMed]
25. Castiglioni, P.; Rienzo, M.D. How the threshold "r" influences approximate entropy analysis of heart-rate variability. In Proceedings of the 2008 Computers in Cardiology, Bologna, Italy, 14–17 September 2008; pp. 561–564.
26. Zhao, L.; Wei, S.; Zhang, C.; Zhang, Y.; Jiang, X.; Liu, F.; Liu, C. Determination of Sample Entropy and Fuzzy Measure Entropy Parameters for Distinguishing Congestive Heart Failure from Normal Sinus Rhythm Subjects. *Entropy* **2015**, *17*, 6270–6288. [CrossRef]

27. Martinez-Mendez, R.; Sekine, M.; Tamura, T. Postural sway parameters using a triaxial accelerometer: Comparing elderly and young healthy adults. *Comput. Methods Biomech. Biomed. Eng.* **2012**, *15*, 899–910. [CrossRef] [PubMed]
28. Kavanagh, J.J. Lower trunk motion and speed-dependence during walking. *J. Neuroeng. Rehabil.* **2009**, *6*, 9. [CrossRef]

sensors

Article

Wearable Multi-Sensor Positioning Prototype for Rowing Technique Evaluation

Luis Rodriguez Mendoza * and Kyle O'Keefe

Geomatics Department, Schulich School of Engineering, University of Calgary, 2500 University Drive NW, Calgary, AB T2N 1N4, Canada; kpgokeef@ucalgary.ca
* Correspondence: luis.alrodmend@gmail.com

Abstract: The goal of this study is to determine the feasibility of a wearable multi-sensor positioning prototype to be used as a training tool to evaluate rowing technique and to determine the positioning accuracy using multiple mathematical models and estimation methods. The wearable device consists of an inertial measurement unit (IMU), an ultra-wideband (UWB) transceiver, and a global navigation satellite system (GNSS) receiver. An experiment on a rowing shell was conducted to evaluate the performance of the system on a rower's wrist, against a centimeter-level GNSS reference trajectory. This experiment analyzed the rowing motion in multiple navigation frames and with various positioning methods. The results show that the wearable device prototype is a viable option for rowing technique analysis; the system was able to provide the position, velocity, and attitude of a rower's wrist, with a positioning accuracy ranging between ±0.185 m and ±1.656 m depending on the estimation method.

Keywords: rowing; ultra-wideband (UWB); GNSS; inertial sensors (IMU); inertial navigation system (INS); wearable technology

Citation: Rodriguez Mendoza, L.; O'Keefe, K. Wearable Multi-Sensor Positioning Prototype for Rowing Technique Evaluation. *Sensors* **2024**, *24*, 5280. https://doi.org/10.3390/s24165280

Academic Editors: Manuel Gil-Martín, Rubén San-Segundo and Fernando Fernández-Martínez

Received: 3 July 2024
Revised: 9 August 2024
Accepted: 13 August 2024
Published: 15 August 2024

1. Introduction

The objective of most competitive rowing is to complete a 2000 m course in the fastest time. Each stroke must be efficient, and each phase must be executed to perfection. The stroke is a movement that goes through four main phases, the catch, the drive, the finish, and the recovery. Rowing technique has been widely researched and evaluated based on the rowing stroke [1–5]. Baudouin and Hawkins concluded that the propulsive force, that directly affects boat velocity, occurs at the oar blade, which is affected by the force and movement of the rower at the handle [5]. Therefore, looking at the oar handle neglects the athlete's anthropometrics and analyzes only the quality of the stroke. Past research has demonstrated that oar handle kinematics correlate with the rower's technique and skill level [1,4]. Handle kinematics have been measured in terms of stroke length, stroke rate, handle velocity, and handle acceleration [4–11]. However, handle positioning is an area that has not been widely explored for estimating these metrics. Additionally, in crew boats, handle position is of great relevance because it can be used to determine the synchrony of the athletes [6,12–14].

Inertial sensors are the most common type of devices used to measure the kinematics of handles in rowing [2,7]. A triad of inertial sensors coupled with an estimator is called an inertial navigation system (INS) [15]. In the context of sport, an INS is often used to estimate kinematic parameters that represent the orientation of a body segment or sport equipment in an inertial frame [16]. Therefore, using an INS to track the motion of the oar handle during the rowing motion is an appropriate approach. However, an INS solution degrades with time due to sensor biases and process errors because it integrates accelerations and angular rates to determine velocity, position, and attitude [15]. Specifically for rowing, one study used inertial and surface electro-myography sensor networks placed on the body of an athlete to identify the muscle activity and acceleration at each stroke phase [6]. This method yielded an understanding of muscle recruitment sequencing and their correlation with the stroke cycle [6]. Moreover, it allowed for athlete synchronization analysis in crew boats [6]. However, this depends on sensor placement and rower's anthropometrics and does not evaluate the motion of the handle. Oar kinematics provide important information about rowers' technique. Reference [4] developed a sensor network consisting of three IMUs; two of these sensors were placed on the oars near to the oarlock, and the last one was placed at the middle of the boat in front of the rower. Reference [4] assessed the technique of 18 rowers based on stroke rate, stroke length, recovery/drive ratio, and feathered/squared blade ratio. Most of the metrics for the results are based on the oar angles measured by the inertial sensors. The results demonstrated that IMUs can be used to obtain important information about oar kinematics. The same research group published two other studies relevant to rowing technique analysis based on inertial sensors [10,11]. In [10], a mobile phone was strapped onto an oar to evaluate stroke length based on the oar angle measured on the phone's IMU and comparing the oar angle to a reference trajectory obtained from a potentiometer [10]. Furthermore, in [11], an IMU was placed inside an oar (instead of a mobile phone) and measured the oar orientation throughout a stroke using an integration algorithm; the model was validated with a reference trajectory obtained from a potentiometer located on the oarlock. This series of studies revealed that inertial sensors can be used in many ways to evaluate rowing technique, especially when placed on the oar. However, none of these papers evaluated the position of the handle, showing a gap in the literature that this article will address. For applications in pedestrian navigation (i.e., running and jogging), pedestrian dead reckoning (PDR) is a method that has been used to improve the navigation performance of low-cost INS based on the periodic motion of the human gait [15]. In addition to PDR, INS errors are typically controlled with external aiding sources of position and velocity, such as GNSS and UWB [17]. Moreover, PDR has been adapted to sports with periodic motions such as cycling [18]; thus, a similar approach is explored for rowing in this study called rowing dead reckoning (RDR).

Moreover, in rowing, radio-frequency-based sensors, such as GNSS and UWB, have not been implemented in technique analysis nor used to control INS errors [2]. GNSS has been used only to track the position and velocity of boats during competitions and training [19–21]. Differential GNSS (DGNSS) has been used as a reference trajectory to validate positioning systems and models [22,23]. Carrier-phase DGNSS can achieve centimeter-level accuracy in open-sky conditions [15], thus making it an appropriate method for validating positioning results. Ultra-wideband (UWB) ranging is a method used for indoor and outdoor localization. Double sided two-way ranging (DS-TWR) is a process that requires the exchange of four messages between a tag and an anchor to estimate the time of arrival (TOA) between messages. Using this process avoids the need for clock and frequency synchronization between tag and anchor [24]. In this paper, a DS-TWR-TOA method is used to obtain distance measurements between two anchors and a single tag. Our previous work studied the validity of UWB to track handle motion indoors on a rowing machine [24]. The system achieved an accuracy of ± 0.21m using a periodic extended Kalman filter (PEKF) in a two-dimensional frame [24]. This paper further explores the system and the model in an outdoor setting and expands the solution to a three-dimensional frame. This study determines whether a wearable positioning system prototype (WPP) that uses UWB ranging, inertial sensors, and a GNSS receiver is a viable tool for rowing technique analysis based on the positioning of the oar handle during the rowing motion. This paper proposes and compares five solutions: UWB, standalone INS, RDR INS, GNSS-aided INS, and UWB-aided INS using a carrier-phase DGNSS as a reference.

The goal of this study is to show that the WPP can fill the gap in the literature by providing information of the position of the oar handle, while demonstrating that radio-frequency technologies such as UWB and GNSS can be used as an accurate source of technique evaluation in rowing. The remainder of this paper is organized as follows: Section 2 describes the WPP and its specifications and covers the mathematical models and algorithms implemented to obtain the handle position in three dimensions. Section 3 shows the evaluation and results using each of the methods, and Section 4 provides conclusions and recommendations.

2. Materials and Methods

2.1. System Description

The WPP consists of two sensor modules, one to access the output of the GNSS data and the other for the UWB and inertial measurement unit (IMU) measurements. The data are transferred in parallel to a logging personal computer (PC). The entire system is powered by a 5V USB power bank. Figure 1 shows the WPP's components ((a) all components, (b) components ready to wear, (c) components on rower) and Table 1 shows the sensor specifications.

Figure 1. Wearable positioning prototype components.

Table 1. Sensor specifications of positioning system.

Sensor	Manufacturer	Model	Description	Data Rate	Accuracy
UWB Receiver	Decawave	DW1000	Single-chip wireless transceiver	50 Hz	±10 cm
MEMS IMU	InvenSense	MPU6050	3-axis gyroscope + 3-axis accelerometer	50 Hz	Accel.: 4 g Gyro.: 500 deg/s
GNSS Receiver	U-Blox	ZED-F9P	Multi-band high precision GNSS module	25 Hz	RTK mode: ±0.01 m + 1 ppm CEP
GNSS Antenna	U-Blox	ANN-MB1	Multiband L1/L5	25 Hz	N/A

The microcontrollers used for the sensor modules are a C099-F9P and a NUCLEO-F446RE [25,26]. They were selected because the C099-F9P was specifically designed by *U-Blox* to easily access GNSS data from the ZED-F9P chip [27] while the NUCLEO-F446RE has available connections which allow for high data rate transmission between UWB radios and reception of IMU measurements simultaneously.

A *Raspberry Pi* (RPi) with a Pimoroni Explorer Hat (PEH) was selected as the logging computer due to its compactness and low power requirements. The RPi is controlled using *VNC Viewer*, which is a remote desktop application. The purpose of using a remote desktop is to eliminate additional hardware (i.e., a screen, mouse, and keyboard) and easy monitoring of the logging process while the user is wearing the WPP. The addition of the PEH is to allow the user to start and stop the logging functions with touch buttons [28].

The output is two log files, one for each module. The UWB/IMU file is a text file with the information of the source (i.e., transmitter ID), the distance observations from the UWB ranging, and the IMU measurements. The GNSS module outputs a customized binary format that can be transformed into a receiver independent exchange (RINEX) format and then processed using the open-source software RTKLIB version 2.4.2 [29].

2.2. Methods

Oar handle kinematics can be described in terms of position, velocity, and attitude angles, expressed in a three-dimensional space. The combination of these terms is often referred to as a navigation solution [30]; it is possible to represent this solution in different coordinate frames. This section introduces the navigation frames used in this study, an overview of attitude angles and coordinate transformations, the mathematical models for processing ultra-wideband ranging measurements, three algorithms used for inertial navigation, and the experimental setup. Figure 2a,b show the boat setup schematic showing the placement of the components with respect to the boat coordinate system.

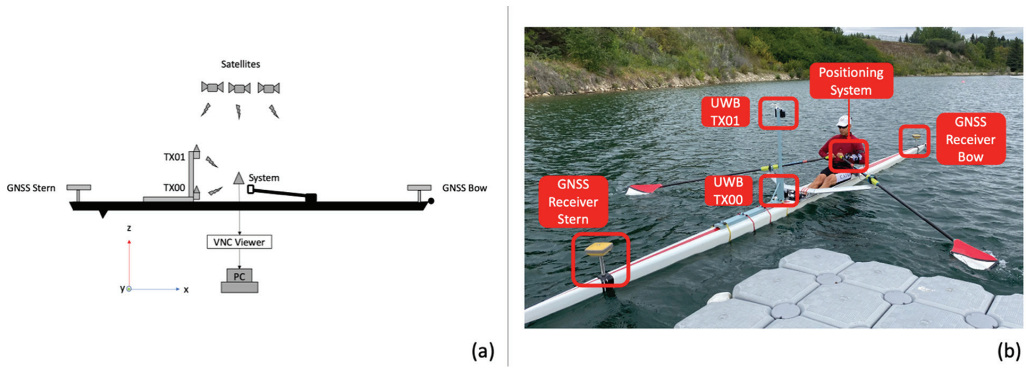


Figure 2. (**a**) Boat setup schematic. (**b**) Boat setup on water.

2.2.1. Body Frame

The sensor or body frame aligns with the axes of the moving object [31]. This study assumes that the origin of the sensor frame (GNSS antenna, UWB antenna, and IMU origin) coincides with the center of gravity of the sensor device which in this case is mounted to the wrist of the rower. This is represented with the superscript and the subscript denoted as $[\blacksquare]_b^b$.

East-North-Up (ENU)

The ENU frame is a local-level or navigation frame [31]. The general representation of a navigation frame is described with the superscript and the subscript denoted as $[\blacksquare]_n^n$. In this paper, ENU uses the World Geodetic System (WGS84) as the reference model for the Earth and has its origin determined by the location of a GNSS base station a few hundred meters away. The coordinate system has the x-axis pointing in the direction of the east, the y-axis towards the true north, and the z-axis in the up direction.

Boat Frame (XYZ)

The boat frame has an origin determined in front of the rower on the rowing shell. The boat frame has the x-axis pointing in the direction along the boat towards the bow (back of the rower), the y-axis towards the starboard (left of the rower), and the z-axis in the vertical direction.

For inertial navigation, it is important to consider the properties of the coordinate systems. The ENU frame is a quasi-inertial frame that allows for inertial navigation. However, the XYZ frame is not inertial; this is because the boat is neither stationary nor moving at a constant speed and instead accelerates and decelerates during different phases of each stroke. Therefore, in order to obtain an inertial navigation solution, this paper determined a boat frame that is instantaneously coincidental with the ENU frame at a moment where it is assumed the rower and oar are momentarily not moving (i.e., translating or rotating), thus creating a static frame for inertial navigation per stroke.

Attitude Angles and Coordinate Transformations

It is possible to transform one system to another one by carrying out a rotation about each of the three rotational axes [31].

Roll (ϕ) is the angle between the body's y-axis (y^b) and the horizontal plane.

Pitch (θ) is the angle between the body's x-axis (x^b) with the horizontal plane.

Azimuth, yaw, or heading (ψ) is the difference between the forward axis with respect to the north in ENU or the along track for the boat frame. Azimuth also represents the rotation angle about the body's z-axis (z^b). The three terms are used interchangeably in this paper.

The relationship between the navigation frame and the body frame can be described with a transformation matrix R_n^b. The rotation sequence in this paper is pitch-roll-azimuth [32]. To do the opposite transformation, the transpose of the rotation matrix is used [30,31].

$$R_n^b = \begin{bmatrix} \cos\psi\cos\theta - \sin\psi\sin\theta\sin\phi & \sin\psi\cos\theta + \cos\psi\sin\theta\sin\phi & -\sin\theta\cos\phi \\ -\sin\psi\cos\phi & \cos\psi\cos\phi & \sin\phi \\ \cos\psi\sin\theta + \sin\psi\sin\phi\cos\theta & \sin\psi\sin\theta - \cos\psi\sin\phi\cos\theta & \cos\theta\cos\phi \end{bmatrix} \tag{1}$$

2.2.2. UWB Positioning Models

The models presented in this study are based on parametric least squares (LS) and the extended Kalman filter (EKF). These models include trilateration, periodic least squares (PLS), constant velocity EKF, and periodic EKF (PEKF). These models were validated for indoor rowing handle tracking in our previous work [24].

Importantly, trilateration gives a single position estimate per epoch that is used as the input of PLS, constant velocity EKF, and PEKF. Moreover, both the constant velocity EKF and PEKF are filtered versions of the trilateration model; however, PEKF uses the solution from PLS as an initialization method.

Trilateration

The WPP receives two UWB ranging measurements at every epoch, one from each transmitter placed in front of the rower. From the boat setup and geometry shown in Figure 2a, having two observations and two unknowns gives a unique two-dimensional solution (x- and z-axis). However, the motion is a three-dimensional movement. Therefore, a range constraint (constant distance measurement) from the oarlock to the WPP is added to provide a third observation and obtain a three-dimensional solution.

Each range measurement can be expressed as a function of the known and unknown positions:

$$f(x) = \sqrt{\left(x_i^{tx} - x^{rx}\right)^2 + \left(y_i^{tx} - y^{rx}\right)^2 + \left(z_i^{tx} - z^{rx}\right)^2} \tag{2}$$

where x_i^{tx}, y_i^{tx}, and z_i^{tx} are the X-, Y-, and Z-coordinates of the transmitter (or oarlock) i, respectively, and x_i^{rx}, y_i^{rx}, and z_i^{rx} are the unknown coordinates of the WPP.

The geometry of the WPP and the transmitters contributes to the accuracy and precision of the trilateration solution, and each set of estimated coordinates is unique. Our previous work showed that the system achieved an accuracy of ± 0.21 m in a two-dimensional frame [24].

Periodic Least Squares (PLS)

PLS is a nonlinear model with a function $f(x,t)$ that describes a periodic wave that models the handle motion during the rowing stroke. This model is based on the work of [33–35] and receives the position estimates of trilateration as input. The number of measurements that represent a stroke cycle varies depending on the stroke rate. A fixed number is manually selected for the first stroke cycle of each test.

The function returns a specific waveform for each axis, and it can be represented in polar coordinates as

$$f(x,t) = A_0 + A_1\cos(\omega t + \phi_1) + A_2\cos(2\omega t + \phi_2) \tag{3}$$

where A_0 is the direct current (DC) offset, A_1 is the first harmonic amplitude, ω is the angular frequency, t is the time, ϕ_1 is the first harmonic phase angle, A_2 is the second harmonic amplitude, and ϕ_2 is the second harmonic phase angle. Thus, the parameter vector ($\hat{x}$) becomes

$$\hat{x} = [A_0\ A_1 A_2 \phi_1\ \phi_2\ \omega]^T \tag{4}$$

The Jacobian matrix of (5) forms the design matrix H_{k+1} and can be written at time k+1 as

$$\begin{gathered} H_{k+1,1} = 1 \\ H_{k+1,2} = \cos(\omega t_k + \phi_1) \\ H_{k+1,3} = \cos(2\omega t_k + \phi_2) \\ H_{k+1,4} = -A_1\sin(\omega t_k + \phi_2) \\ H_{k+1,5} = A_2\sin(2\omega t_k + \phi_2) \\ H_{k+1,6} = -A_1 t_k \sin(\omega t_k + \phi_1) - 2A_2 t_k \sin(2\omega t_k + \phi_2) \end{gathered} \tag{5}$$

This method estimates the position of the handle throughout a full cycle (stroke) independently for each axis (i.e., each axis has independent parameter vectors including frequency).

Extended Kalman Filter (EKF) Overview

EKF is an iterative estimation method for nonlinear functions that has been widely used in navigation. EKF extends the LS estimation with the prediction of the state vector, commonly known as the dynamic model [34–37]. The Kalman filter loop is shown in Figure 3.

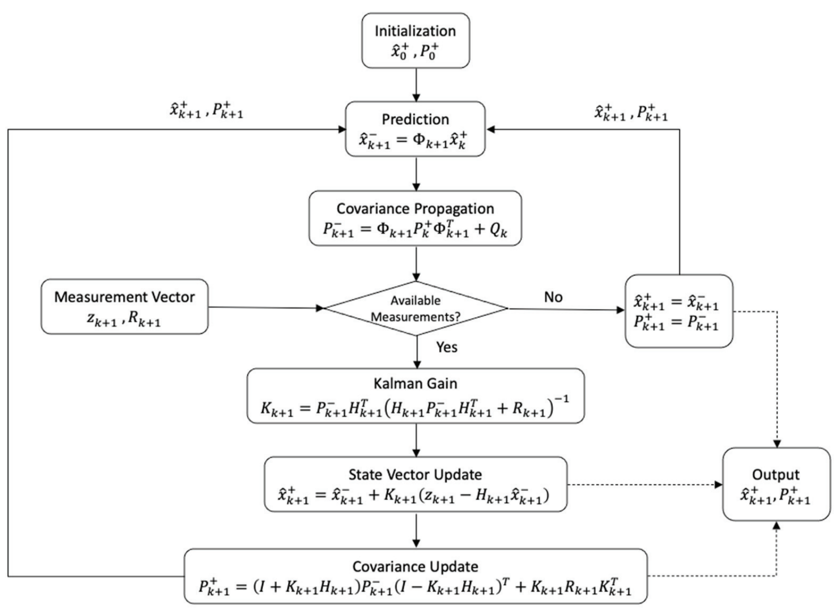


Figure 3. Discrete Kalman filter flowchart.

Constant Velocity EKF

The EKF with constant velocity assumes that the object being tracked is moving at a constant speed. The state vector is formed by two parameters, position (x, y, z) and velocity (v_x, v_y, and v_z):

$$\hat{x}_k = \begin{bmatrix} x \; y \; z \; v_x \; v_y \; v_z \end{bmatrix}^T \tag{6}$$

In the dynamic model, the state transition matrix Φ_{k+1} shows the relationship between time and distance to predict the velocity and position of the handle at time k:

$$\Phi_{k+1} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

The time between epochs is determined as $\Delta t = t_k - t_{k+1}$. When the measurement vector z_{k+1} is also formed by position and velocity, the design matrix H_{k+1} is a 6×6 identity matrix.

Periodic EKF (PEKF)

PEKF is an extended version of PLS that includes a prediction in the algorithm. The filter is initialized with PLS for fast convergence. After the filter is initialized, the measurements from trilateration are the input.

The state vector $\hat{x}$ and the design matrix H_{k+1} in PEKF are the same as in Equations (4) and (6), and the transition matrix Φ_{k+1} is a 6×6 identity matrix. This dynamic model is selected because of the assumption that the best prediction is that the next stroke is similar to the previous stroke. This assumption is vulnerable to changes in frequency, but the addition of process noise allows the filter to place more weight on the new set of observations than on the prediction from the previous stroke.

2.2.3. Strapdown Inertial Navigation System (INS)

An INS has three main components: an IMU, a pre-processing unit, and a mechanization module [31]. In strapdown systems, the sensors are rigidly mounted onto the body of the moving object. This study uses a low-cost micro-electromechanical system (MEMS) IMU formed by two tri-axial sensors: an accelerometer and a gyroscope. The mechanization module consists of a series of differential navigation equations that can be written in the local-level frame as follows [30,31]:

$$\begin{pmatrix} \dot{r}^l \\ \dot{V}^l \\ \dot{R}_b^l \end{pmatrix} = \begin{pmatrix} V^l \\ R_b^l f^b + g^l \\ R_b^l \Omega_{ib}^b \end{pmatrix} \tag{8}$$

where $[\dot{r}^l, \dot{V}^l, \dot{R}_b^l]^T$ are the time derivatives of the navigation states: position, velocity, and attitude [31]. f^b and Ω_{ib}^b are the specific force and skew-symmetric matrix of the angular rate measurements (ω^b), respectively. V^l is the velocity vector and g^l is the normal gravity vector.

The mechanization module is initialized with a set of measurements (f^b and ω^b) and an initial alignment using attitude angles. The initial roll and pitch can be calculated using the accelerometer measurements when the device is at rest, with the following equations:

$$\phi = \tan^{-1}\left(\frac{f_y^b}{\sqrt{f_x^{b2} + f_z^{b2}}}\right) \tag{9}$$

$$\theta = \tan^{-1}\left(\frac{-f_x^b}{f_z^b}\right) \tag{10}$$

In this study, the azimuth is defined by the user because the MEMS-grade IMU does not have the ability to provide an orientation. A step-by-step description of the mechanization process can be found in [31].

2.2.4. INS Loosely Coupled Integration

Loosely coupled integration is a cascade architecture where an aiding source is used to estimate sensor biases and errors from the mechanization process by introducing a Kalman filter [15,17,37,38].

The output of the loosely coupled integration is the corrected inertial navigation solution (i.e., position, velocity, and attitude) and the estimates of the sensor biases. This paper uses a closed-loop error-state Kalman filter algorithm. Figure 4 illustrates the loosely coupled integration process. The Kalman filter state vector can be described as

$$\delta x = \begin{bmatrix} \delta r_{eb}^n \\ \delta V_{eb}^n \\ \delta \Psi_{nb}^n \\ b_f \\ b_\omega \end{bmatrix} \tag{11}$$

where b_f and b_ω are the accelerometer and gyroscope biases, respectively. It is assumed that the bias states are modeled as first-order Gauss-Markov processes [31].

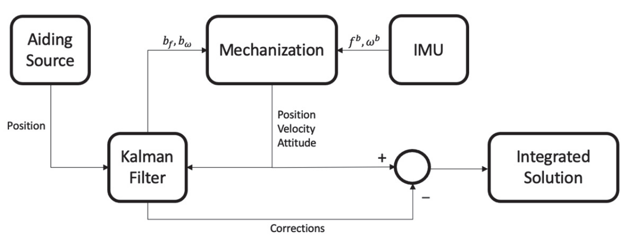


Figure 4. Loosely coupled integration flowchart.

The state transition matrix can be written as

$$\Phi_{k+1} = \begin{pmatrix} I_{3\times3} & I_{3\times3}\Delta t & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & -\Omega\left(\hat{f}_{ib}^n\right)\Delta t & \hat{R}_b^n\Delta t & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & 0_{3\times3} & \hat{R}_b^n\Delta t \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} - \beta_{b_f}\Delta t & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} - \beta_{b_\omega}\Delta t \end{pmatrix} \tag{12}$$

where β_{b_f} and β_{b_ω} are the diagonal matrices of the inverse of the correlation time (τ) for the sensor biases.

$$\beta_{b_f} = \begin{pmatrix} \frac{1}{\tau_{b_{fx}}} & 0 & 0 \\ 0 & \frac{1}{\tau_{b_{fy}}} & 0 \\ 0 & 0 & \frac{1}{\tau_{b_{fz}}} \end{pmatrix} \text{ and } \beta_{b_\omega} = \begin{pmatrix} \frac{1}{\tau_{b_{\omega x}}} & 0 & 0 \\ 0 & \frac{1}{\tau_{b_{\omega y}}} & 0 \\ 0 & 0 & \frac{1}{\tau_{b_{\omega z}}} \end{pmatrix} \tag{13}$$

The measurement vector is the difference between the INS and the aiding source positions:

$$z_{k+1} = r_{INS} - r_{Aiding} = \delta r \tag{14}$$

And the design matrix H_{k+1} can be defined as follows:

$$H_{k+1} = [I_{3\times3}\ 0_{3\times3}\ 0_{3\times3}\ 0_{3\times3}\ 0_{3\times3}] \tag{15}$$

2.2.5. Rowing Dead Reckoning (RDR)

We propose an analogous model to PDR to estimate the kinematics of oar handles that we call rowing dead reckoning (RDR). The first step of RDR is to identify a finish position (new stroke).

At the finish position, the magnitude of the acceleration in the accelerometer's x-axis is the largest (because the propulsion has ended and the boat is not decelerating). Similarly, the angular velocity on the gyroscope's z-axis shows a well-defined peak (since the finish corresponds to a change in direction of the oar). In Figure 5, the two peaks are highlighted to show their correlation. This information is used to determine a new stroke and reset the initial conditions of the mechanization equations to minimize the accumulation of errors.

Figure 5. X accelerometer and Z gyroscope raw data.

At the finish, it is assumed that the rower and oar are momentarily not moving (i.e., translating or rotating) in an instantaneous XYZ frame defined for the next stroke that coincides and aligns with the ENU frame at that instant.

RDR Stroke Detection Algorithm:

1. **procedure** StrokeDetection (time (t), accelerometer (f_{ib}^{b}), gyroscope (ω_{ib}^{b}))
2. Define threshold for acceleration magnitude, T_f
3. Define threshold for angular velocity, T_w
4. Choose sliding window size, N
5. Calculate mean values at time (t), $\overline{f_{ib}^{b}}(t)$, $\overline{\omega_{ib}^{b}}(t)$
6. **If** { $\overline{f_{ib}^{b}}(t) > T_f$ ***and*** $\overline{\omega_{ib}^{b}}(t) > T_\omega$ ***and*** $\overline{f_{ib}^{b}}(t) > \overline{f_{ib}^{b}}(t-1)$} **then**
7. **If** { $t - (t-1) < 1s$} **then**
8. **continue**
9. End **if**
10. Declare new stroke at time (t)
11. End **if**
12. End **procedure**

The RDR stroke detection algorithm requires one to compute the mean values of sensor readings to reduce noise and prevent false stroke detections. Additionally, a time constraint of one-second separations between strokes is applied to prevent false detections. Then, mechanization equations from strapdown INS are applied. The orientation of the IMU at the instant that the recovery is detected, set to assumed values.

The next section describes the experimental setup for testing the WPP.

2.2.6. Experiment

A test of the WPP was conducted at the Victoria City Rowing Club (VCRC) using a *Hudson* single scull, three *Topcon HiPer SR* multi-band GNSS receivers, and the proposed WPP. In Figure 2a,b, the U-blox GNSS receiver in the WPP serves both as an example of a low-cost wearable GNSS device (using single-frequency pseudo-ranges only) and a source for the reference trajectory (using dual-frequency carrier-phase observations).

Two of the *Topcon* receivers are located on the stern and on the bow, and the third is used as a base station for post-processing with a sampling rate of 10 Hz. The stroke rate maintained for this experiment was between 18 and 36 strokes per minute, meaning a maximum frequency of 0.6 Hz, well below the frequency of the sampling frequency of the reference trajectory. In terms of the UWB radios and the U-blox receiver, the sampling frequency was 50 Hz and 25 Hz, respectively.

The output of these receivers established the absolute position and orientation of the rowing shell and the transformation between the navigation and body frame of the shell. *RTKLIB* was used to post-process the data in carrier-phase DGNSS to obtain centimeter-level accuracy.

The carrier-phase DGNSS solution obtained in post-processing was used as the reference trajectory to evaluate the results obtained from the models presented in this paper. The expected accuracy from the *Topcon* receivers for DGNSS was 10 mm +0.8 ppm and 15 mm + 1.0 ppm for the horizontal and vertical axes (1σ), respectively [39]. For the WPP, the *U-blox* receiver specified an accuracy of 12 mm +1.2 ppm (1σ) for both the horizontal and vertical axes, also in carrier-phase DGNSS mode [40].

The UWB transmitters were placed on a customized stand facing the user, and the WPP was attached to the user's left wrist and waist.

Testing included rowing in various directions, speeds, stroke rates, and stroke lengths for approximately 90 min. Figure 6 shows the trajectory of the boat for the 90 min; from this figure, it is possible to observe that the trajectory was not in straight lines nor in the same location (i.e., back and forth). The two "small" loops on the eastern side of the lake were rowed in a clockwise direction, while the "large" loop on the western side of the lake was rowed in a counterclockwise direction. The speed of the boat and stroke length varied based on the side of the lake (i.e., water conditions), wind speed, and stroke rate. Figure 7 shows a subset of the test that was used to validate the models presented in this paper. This window was selected because the trajectory includes a section of rowing at a constant stroke rate, one section with an increase in rate (higher frequency), and one section with a decrease in rate (lower frequency). Additionally, the trajectory included a slight turn to the southwest. Recall that the purpose of this paper is to determine the feasibility of the WPP to be used as a technique analysis tool by providing information on the position of the handle based on UWB, GNSS, and INS with various mathematical models. The intention of this test was to evaluate common rowing patterns in a regular training session.

The next section presents the results obtained from the experimentation using the models shown in this section.

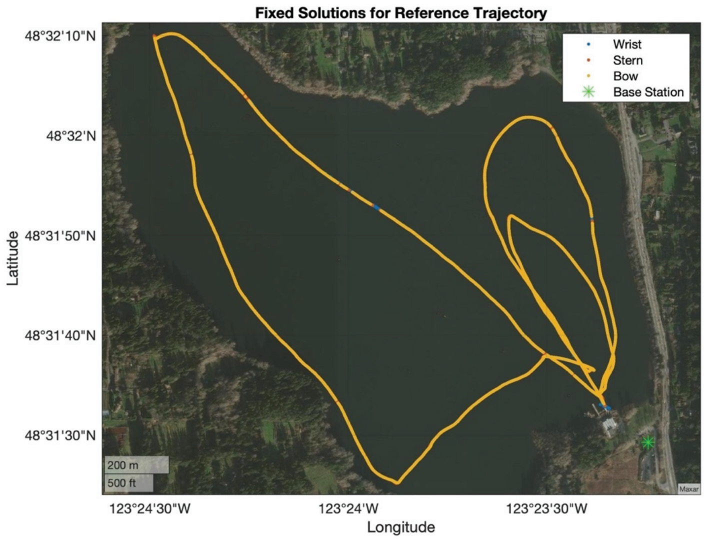


Figure 6. Experiment trajectory.

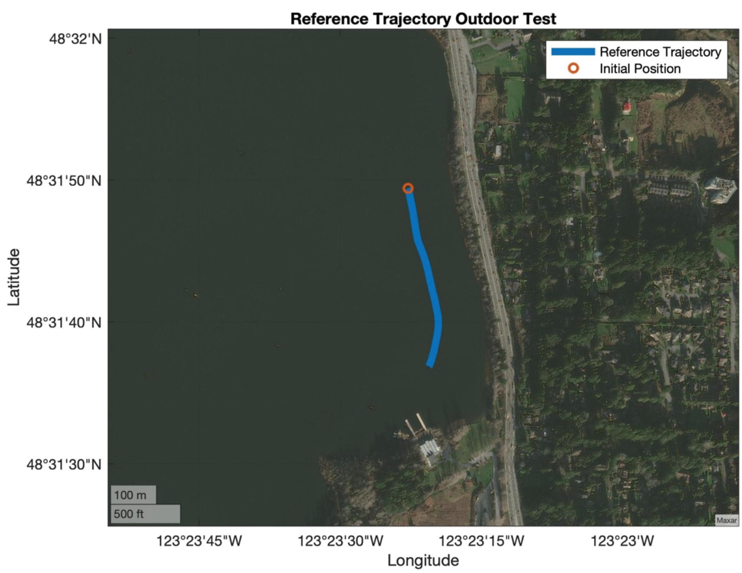


Figure 7. Detailed evaluation window.

3. Results and Discussion

This section presents the results with each positioning model using a 120 s segment of the data that includes changes in direction, speed, stroke rate, and stroke length.

3.1. UWB Results

The accuracy of the models, ranges (TX00 $\pm$ 0.121 m, TX01 $\pm$ 0.106 m), and range constraint ($\pm$0.018 m) is evaluated using the GNSS reference trajectory. Figure 8 shows the position estimates for the X-, Y-, and Z-coordinates, respectively. Figure 8a–c contains the coordinate at the top and the error with respect to the reference at the bottom. The reference trajectory is represented with a green line, the trilateration is represented with red dots, the

EKF with constant velocity is represented with yellow dots, the PEKF is represented with purple dots, and the time of update is represented with black triangles.

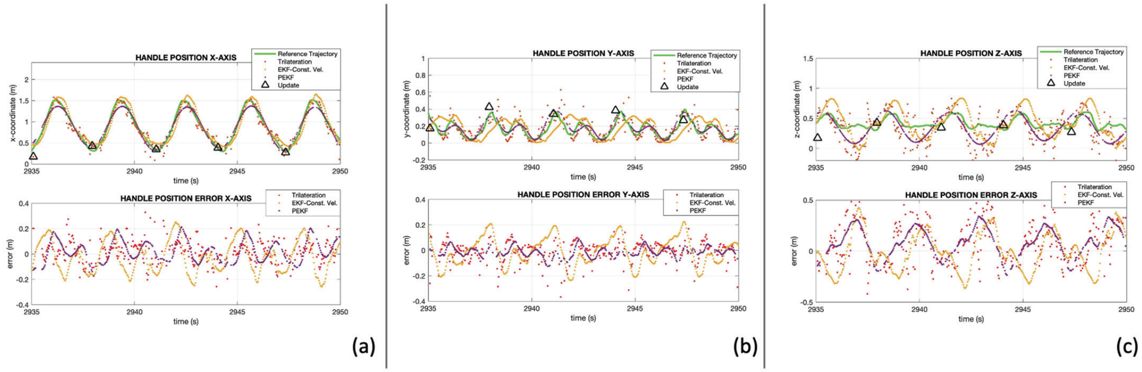


Figure 8. UWB positioning results, (**a**) X-coordinate, (**b**) Y-coordinate, (**c**) Z-coordinate in boat frame.

On the X-coordinate, PEKF reduced the variability of the trilateration solution and represented the handle position very accurately. The error is observed to be below ±0.20 m. The results from trilateration also represented the X-axis accurately but with some variance between estimates. Lastly, EKF with constant velocity overestimated the catch and the finish and returned the solution with the largest errors.

On the Y-axis, the handle motion shows two waves with different amplitudes at each stroke. The larger wave corresponds to the catch and the smaller one corresponds to the finish. PEKF represents both curves accurately. The results from trilateration follow the handle motion with a larger visible variance. EKF with constant velocity is not able to fully represent the two curves in the stroke. This is due to the rapid changes in velocities at the peaks and valleys from each curve. The largest observed errors for this model are within ±0.20 m, which is an acceptable accuracy in terms of positioning but not enough to represent the motion of handles for technique analysis.

On the Z-coordinate, the models can estimate a periodic curve; however, the amplitude is overestimated. This was expected as the accuracy of the ranges is approximately ±0.12 m and the amplitude of the motion is approximately 0.2 m. However, PEKF can be used to estimate the time of the catch, which is valuable for crew coordination and technique analysis.

PEKF outperformed trilateration and EKF with constant velocity. The output from this model clearly shows the motion of the handle and estimates the curves from each stroke on every axis.

3.2. INS Results

The roll and pitch angles used at the finish position are −38.5 deg and −24.9 deg, respectively. These angles were obtained through testing and manual adjustments.

The azimuth is assumed to be −45 deg; thus, computed angles about the Z-axis are with respect to an arbitrary start position rather than with respect to the forward axis of the boat.

The epoch-by-epoch DGNSS reference trajectory was numerically differentiated to obtain velocity and acceleration. The accuracy of the reference, from the carrier-phase DGNSS specifications, is an order or magnitude better the inertial solution and not affected by bias or drift.

Figure 9a shows the resulting acceleration of the oar handle in the instantaneous inertial XYZ frame. The acceleration is well aligned at the beginning of the test. However, it begins to drift over time. Some drift was expected due to the sensor biases; however, the bias effect on the attitude (Figure 9d) also influences the acceleration, thus affecting the velocity and position estimates.

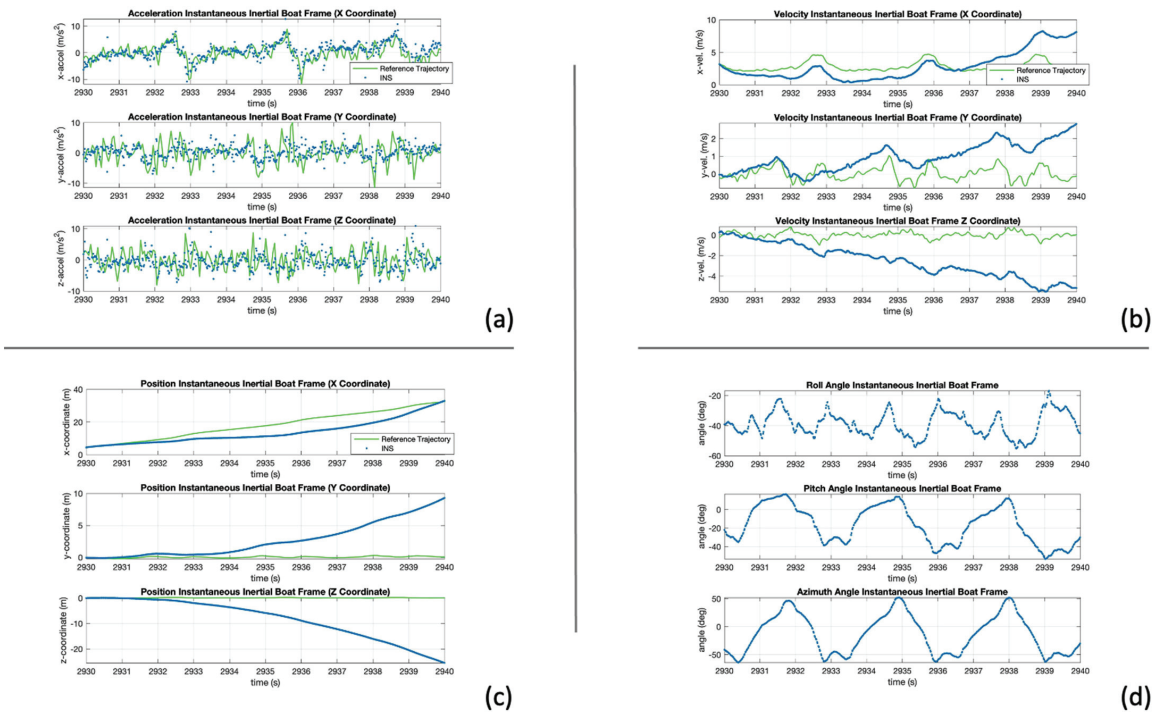


Figure 9. INS strapdown results, (**a**) acceleration, (**b**) velocity, (**c**) position, (**d**) attitude in instantaneous inertial boat frame.

Figure 9b shows the estimated velocity. All three axes are affected by the biases, especially on the Z-coordinate because some of the acceleration on the horizontal plane is transformed into this axis.

Lastly, Figure 9c shows the estimated position of the handle. The estimates on the horizontal plane show similar characteristics as the reference trajectory and begin to drift after some seconds. On the other hand, the effect of the biases and errors greatly influence the Z-coordinate and none of the expected characteristics are observable.

The results shown above demonstrated that standalone INS is not sufficient to estimate the position of the handle and requires an aiding method to reduce the effects of the sensor biases.

3.3. Rowing Dead Reckoning (RDR)

RDR is expected to reduce drift errors as each stroke is analyzed independently. Figure 10a shows the acceleration and reference trajectory in the instantaneous inertial XYZ frame. This figure shows that all axes align with the reference trajectory, demonstrating that RDR can reduce the drift from the sensor biases. This is confirmed on Figure 10d; the attitude angles are reset at every stroke, reducing the drift.

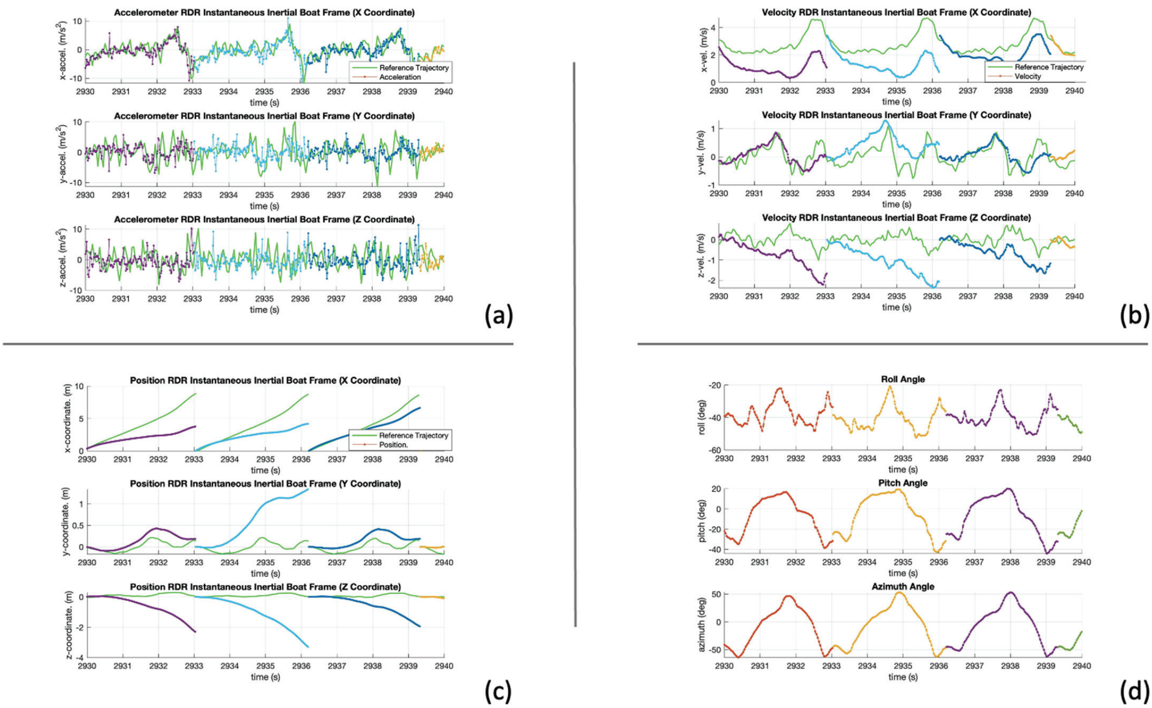


Figure 10. INS RDR results, (**a**) acceleration, (**b**) velocity, (**c**) position, (**d**) attitude in instantaneous inertial boat frame.

Figure 10b shows the velocity, some drift remains affecting all three axes, especially the X- and Z-coordinates. RDR constraints the solution and limits the effect of the biases and errors. However, these biases and errors are not estimated or corrected.

Lastly, the position estimates on Figure 10c show that RDR improves the accuracy of standalone INS without any additional hardware.

3.4. INS/GNSS Integration Results

This section shows the positioning results of the integrated solution from INS and standalone GNSS. On Figure 11, the positions from the standalone GNSS are shown in red, the reference trajectory is in green, and the INS/GNSS solution is in yellow.

First, Figure 11a–c show the position of the handle in ENU. The accuracy in the horizontal plane is below the meter level, which is excellent for navigation. However, the vertical plane is at the meter level. The ENU frame does not provide information about the technique of the rower. Therefore, a transformation into the XYZ frame is needed. Figure 11d–f show the resulting positions. The X- and Y-coordinates provide an estimation of the motion of the handle. However, the accuracy is not sufficient to assess technique. Furthermore, due to the meter-level accuracy of the Z-coordinate, the motion of the handle cannot be observed. It should be noted that the transformation from the ENU to the XYZ frame depends on the two reference GNSS receivers attached to the boat, making this method impractical for a casual user wearing only a GNSS/INS smartwatch, for example.

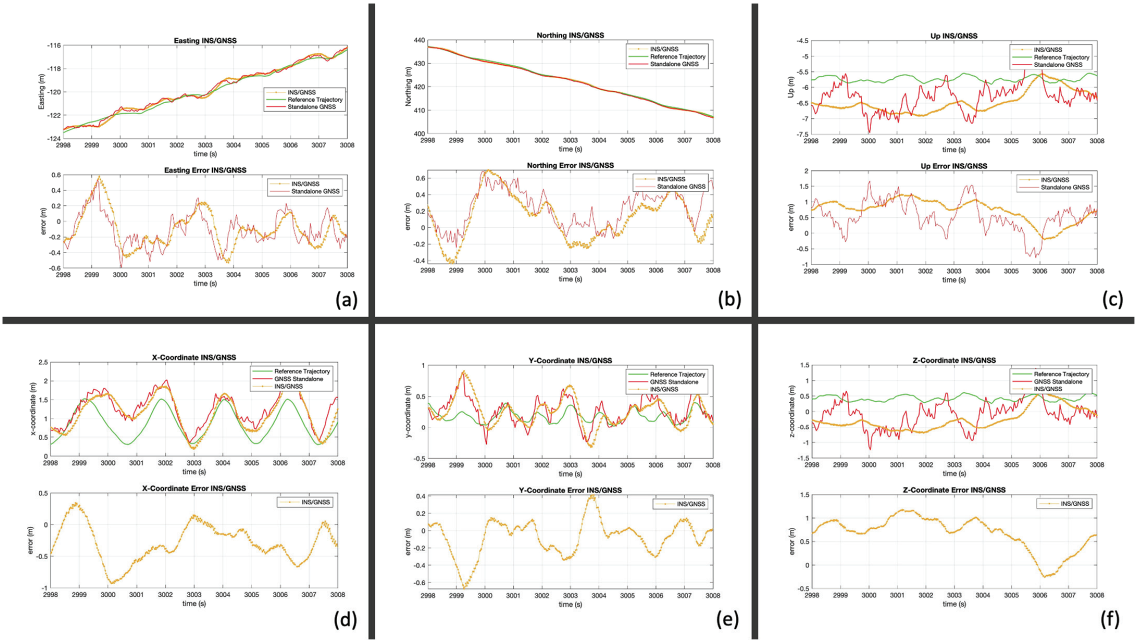


Figure 11. INS/GNSS results. Top ENU frame: (**a**) easting coordinate, (**b**) northing coordinate, (**c**) up coordinate. Bottom boat frame: (**d**) X-coordinate, (**e**) Y-coordinate, (**f**) Z-coordinate.

3.5. INS/UWB Integration Results

Trilateration and PEKF demonstrated to be the best models to represent the motion of the handle. Therefore, these are the two models used in the integration. This section shows the results in the XYZ frame. Figure 12 shows the INS/trilateration integration at the top (a–c) and the INS/PEKF integration at the bottom (d–f).

Both aiding sources greatly improved the INS solution and allowed for an accurate representation of the motion of the handle in all axes. These figures demonstrate that UWB can enhance the solution from INS to estimate biases, correct mechanization errors, and allow for technique analysis. However, this method, similar to INS/GNSS, requires the boat reference, provided by two additional GNSS receivers, to transform between frames.

Tables 1–4 summarize the errors calculated from each method and navigation frame. The error is presented as the mean value and its standard deviation, which represents the accuracy and precision of the solution.

Table 2. Error summary results in boat frame.

3D Outdoor Test Boat Frame Overall (120 s)			
Method	**X-axis**	**Y-axis**	**Z-axis**
	Mean Error-Std. (m)	**Mean Error-Std. (m)**	**Mean Error-Std. (m)**
Trilateration UWB	0.023 ± 0.105	-0.015 ± 0.104	0.048 ± 0.222
PEKF UWB	0.022 ± 0.121	-0.020 ± 0.062	0.045 ± 0.129
EKF Const. Vel. UWB	-0.028 ± 0.152	0.002 ± 0.118	-0.025 ± 0.189
INS/GNSS	-0.250 ± 0.458	-0.060 ± 0.503	1.269 ± 1.207
INS/Trilateration UWB	-0.061 ± 0.101	-0.094 ± 0.127	0.103 ± 0.148
INS/PEKF UWB	-0.070 ± 0.117	-0.096 ± 0.106	0.136 ± 0.104

Table 3. Error summary results in instantaneous boat frame.

3D Outdoor Test Instantaneous Inertial Boat Frame Overall (120 s)			
Method	**X-axis**	**Y-axis**	**Z-axis**
	Mean Error-Std. (m)	**Mean Error-Std. (m)**	**Mean Error-Std. (m)**
INS/RDR	0.172 ± 1.236	−0.583 ± 0.695	0.892 ± 0.856

Table 4. Error summary results in ENU frame.

3D Outdoor Test ENU Frame Overall (120 s)			
Method	**Easting**	**Northing**	**Up**
	Mean Error-Std. (m)	**Mean Error-Std. (m)**	**Mean Error-Std. (m)**
INS/GNSS	−0.077 ± 0.496	0.131 ± 0.433	1.325 ± 1.207
INS/Trilateration UWB	0.088 ± 0.129	0.077 ± 0.091	0.103 ± 0.148
INS/PEKF UWB	0.089 ± 0.105	0.086 ± 0.111	0.136 ± 0.104

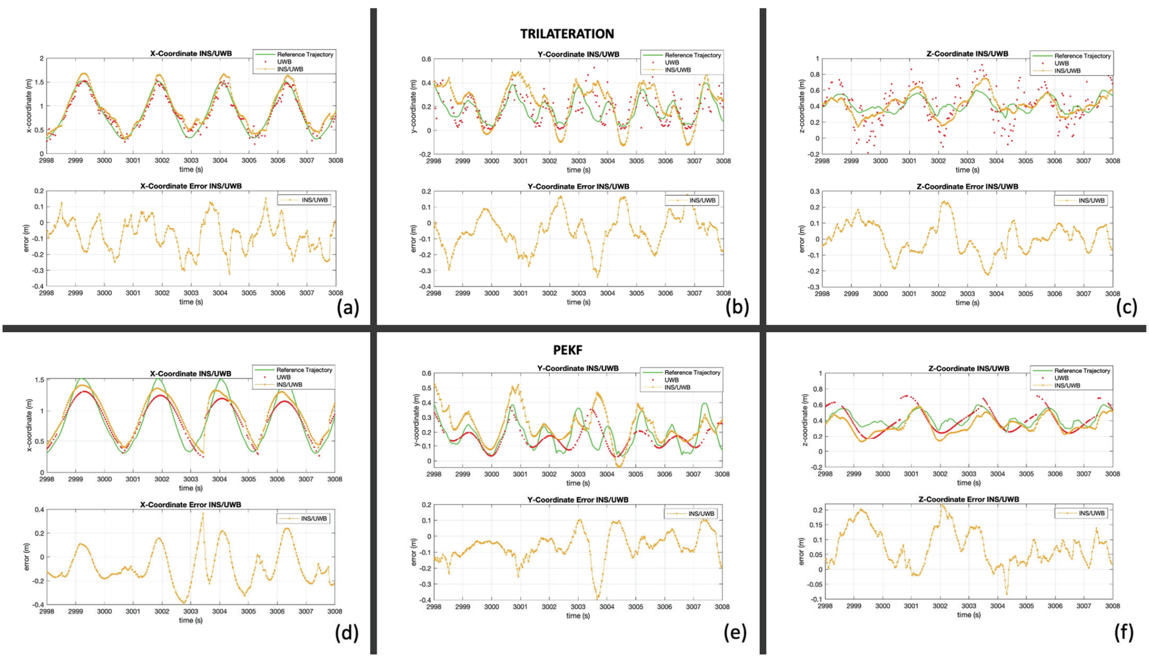


Figure 12. INS/UWB results for boat frame. Top INS/trilateration: (**a**) X-coordinate, (**b**) Y-coordinate, (**c**) Z-coordinate. Bottom INS/PEKF: (**d**) X-coordinate, (**e**) Y-coordinate, (**f**) Z-coordinate.

4. Conclusions

The WPP demonstrated to be a feasible option for rowing technique analyses because it was able to provide position, velocity, and attitude when INS was integrated with UWB or GNSS. Thus, providing a full understanding of the oar/wrist movement.

UWB trilateration, PEKF, and EKF with constant velocity had total accuracies in the boat frame of ±0.267 m, ±0.187 m, and ±0.270 m, respectively, demonstrating that PEKF was the most accurate UWB standalone positioning method. The integrated methods, INS/GNSS, INS/trilateration UWB, and INS/PEKF UWB obtained accuracies of ±1.386 m,

±0.219 m, and ±0.189 m, respectively, in the boat frame, highlighting that an integrated method provides similar accuracy than standalone UWB with the additional information of velocity and attitude from INS.

In the instantaneous inertial boat frame, integrating INS with RDR had a total accuracy of ±1.656 m. This method was demonstrated to have reduced the effects of sensor biases and reduced accumulated errors. However, it was the least accurate of the integrated methods.

Lastly, in the ENU frame, the integrated methods INS/GNSS, INS/trilateration UWB, and INS/PEKF UWB resulted in accuracies of ±1.375 m, ±0.216 m, and ±0.185 m, respectively, showing that the accuracy of the positioning methods was maintained between navigation frames and that it was possible to obtain the position of the moving boat.

Author Contributions: Conceptualization, L.R.M. and K.O.; methodology, L.R.M. and K.O.; software, L.R.M.; validation, L.R.M.; formal analysis, L.R.M.; investigation, L.R.M.; resources K.O.; data curation, L.R.M.; writing—original draft preparation, L.R.M.; writing—review and editing, K.O.; visualization, L.R.M.; supervision, K.O.; project administration, K.O.; funding acquisition, L.R.M. and K.O. All authors have read and agreed to the published version of the manuscript.

Funding: This study was partially funded by NSERC CREATE We-TRAC Project No. CREATE/511166-2018.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Raw data from this study can be obtained by contacting the corresponding author.

Acknowledgments: We would like to thank the Calgary Rowing Club and Victoria City Rowing Club for their support toward this project.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Soper, C.; Hume, P.A. Towards an ideal rowing technique for performance. *Sports Med.* **2004**, *34*, 825–848. [CrossRef] [PubMed]
2. Worsey, M.T.; Espinosa, H.G.; Shepherd, J.B.; Thiel, D.V. A systematic review of performance analysis in rowing using inertial sensors. *Electronics* **2019**, *8*, 1304. [CrossRef]
3. Smith, R.; Loschner, C. Biomechanics feedback for rowing. *J. Sports Sci.* **2002**, *20*, 783–791. [CrossRef] [PubMed]
4. Tessendorf, B.; Gravenhorst, F.; Arnrich, B.; Tröster, G. An imu-based sensor network to continuously monitor rowing technique on the water. In Proceedings of the 2011 Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Adelaide, Australia, 6–9 December 2011; IEEE: New York, NY, USA, 2011; pp. 253–258.
5. Baudouin, A.; Hawkins, D. A biomechanical review of factors affecting rowing performance. *Br. J. Sports Med.* **2002**, *36*, 396–402. [CrossRef]
6. Armstrong, A.; Nokes, L. Sensor node acceleration signatures and electromyography in synchronization and sequencing analysis in sport: A rowing perspective. *Proc. Inst. Mech. Eng. Part P J. Sports Eng. Technol.* **2017**, *231*, 253–261.
7. Ruffaldi, E.; Peppoloni, L.; Filippeschi, A. Sensor fusion for complex articulated body tracking applied in rowing. *Proc. Inst. Mech. Eng. Part P J. Sports Eng. Technol.* **2015**, *229*, 92–102. [CrossRef]
8. Kleshnev, V. Boat acceleration, temporal structure of the stroke cycle, and effectiveness in rowing. *Proc. Inst. Mech. Eng. Part P J. Sports Eng. Technol.* **2010**, *224*, 63–74. [CrossRef]
9. Llosa, J.; Vilajosana, I.; Vilajosana, X.; Navarro, N.; Suriñach, E.; Marquès, J.M. REMOTE, a wireless sensor network based system to monitor rowing performance. *Sensors* **2009**, *9*, 7069–7082. [CrossRef]
10. Gravenhorst, F.; Muaremi, A.; Kottmann, F.; Tröster, G.; Sigrist, R.; Gerig, N.; Draper, C. Strap and row: Rowing technique analysis based on inertial measurement units implemented in mobile phones. In Proceedings of the 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 21–24 April 2014; IEEE: New York, NY, USA, 2014; pp. 1–6.
11. Gravenhorst, F.; Turner, T.; Draper, C.; Smith, R.M.; Troester, G. Validation of a rowing oar angle measurement system based on an inertial measurement unit. In Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Melbourne, VIC, Australia, 16–18 July 2013; IEEE: New York, NY, USA, 2013; pp. 1412–1419.
12. Avvenuti, M.; Cesarini, D.; Cimino, M.G. Mars, a multi-agent system for assessing rowers' coordination via motion-based stigmergy. *Sensors* **2013**, *13*, 12218–12243. [CrossRef]
13. Schaffert, N.; Mattes, K. Influence of acoustic feedback on boat speed and crew synchronization in elite junior rowing. *Int. J. Sports Sci. Coach.* **2016**, *11*, 832–845. [CrossRef]

14. Cuijpers, L.S.; Passos, P.J.M.; Murgia, A.; Hoogerheide, A.; Lemmink, K.A.P.M.; de Poel, H.J. Rocking the boat: Does perfect rowing crew synchronization reduce detrimental boat movements? *Scand. J. Med. Sci. Sports* **2017**, *27*, 1697–1704. [CrossRef] [PubMed]
15. Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed.; Artech House: Boston, MA, USA, 2013.
16. Camomilla, V.; Bergamini, E.; Fantozzi, S.; Vannozzi, G. Trends supporting the in-field use of wearable inertial sensors for sport performance evaluation: A systematic review. *Sensors* **2018**, *18*, 873. [CrossRef] [PubMed]
17. Aggarwal, P.; Syed, Z.; Noureldin, A.; Syed, Z. *MEMS-Based Integrated Navigation*; Artech House: Boston, MA, USA, 2010.
18. Chang, H.W.; Georgy, J.; El-Sheimy, N. Cycling dead reckoning for enhanced portable device navigation on multi-gear bicycles. *GPS Solut.* **2015**, *19*, 611–621. [CrossRef]
19. Dubus, G. Evaluation of four models for the sonification of elite rowing. *J. Multimodal User Interfaces* **2012**, *5*, 143–156. [CrossRef]
20. Groh, B.H.; Reinfelder, S.J.; Streicher, M.N.; Taraben, A.; Eskofier, B.M. Movement prediction in rowing using a Dynamic Time Warping based stroke detection. In Proceedings of the 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 21–24 April 2014; IEEE: New York, NY, USA, 2014; pp. 1–6.
21. Day, A.H.; Campbell, I.; Clelland, D.; Cichowicz, J. An experimental study of unsteady hydrodynamics of a single scull. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2011**, *225*, 282–294. [CrossRef]
22. Cloud, B.; Tarien, B.; Liu, A.; Shedd, T.; Lin, X.; Hubbard, M.; Crawford, R.P.; Moore, J.K. Adaptive smartphone-based sensor fusion for estimating competitive rowing kinematic metrics. *PLoS ONE* **2019**, *14*, 1–27. [CrossRef] [PubMed]
23. Mpimis, A.; Gikas, V. Monitoring and evaluation of rowing performance using mobile mapping data. *Arch. Fotogram. Kartogr. I Teledetekcji* **2011**, *22*, 337–349.
24. Rodriguez Mendoza, L.; O'Keefe, K. Periodic extended kalman filter to estimate rowing motion indoors using a wearable ultra-wideband ranging positioning system. In Proceedings of the 2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Lloret de Mar, Spain, 29 November–2 December 2021; IEEE: New York, NY, USA, 2021; pp. 1–8.
25. U-Blox. C099-F9P: Application Board (Rev. E), ODIN-W2 u-ConnectXpress Software-User Guide. 2020. Available online: https://content.u-blox.com/sites/default/files/C099-F9P-AppBoard-ODIN-W2-uCX_UserGuide_(UBX-18055649).pdf (accessed on 29 June 2021).
26. STMicroelectronics. UM2324: User Manual STM32 Nucleo-64 Boards. 2020. Available online: https://www.st.com/resource/en/user_manual/dm00452640-stm32-nucleo-64-boards-with-stm32g07xrb-mcus-stmicroelectronics.pdf (accessed on 20 March 2021).
27. U-Blox. ZED-F9P-02B: U-Blox F9 High Precision GNSS Module-Data Sheet. 2021. Available online: https://www.verical.com/datasheet/u-blox-gps-receiver-ZED-F9P-02B-7307625.pdf (accessed on 21 June 2021).
28. Pimoroni. GitHub Inc. 2021. Available online: https://github.com/pimoroni/explorer-hat (accessed on 21 June 2021).
29. RTKLIB. RTKLIB ver. 2.4.2. *Manual.* 2023. Available online: https://www.scribd.com/document/494593890/manual-2-4-2 (accessed on 21 June 2021).
30. Tjhai, C. Integration of Multiple Low-Cost Wearable Inertial/Magnetic Sensors and Kinematics of Lower Limbs for Improving Pedestrian Navigation Systems. Ph.D. Thesis, University of Calgary, Calgary, AB, Canada, 2019.
31. Noureldin, A.; Karamat, T.B.; Georgy, J. *Fundamentals of Inertial Navigation, Satellite-Based Positioning and Their Integration*; Springer: Berlin/Heidelberg, Germany, 2013.
32. Pedley, M. Tilt Sensing Using a Three-Axis Accelerometer. Report, Freescale Semiconductor. 2013. Available online: https://www.nxp.com/docs/en/application-note/AN3461.pdf (accessed on 21 June 2021).
33. Uner, M. Frequency, Amplitude, and Phase Tracking of Nonsinusoidal Signal in Noise with Extended Kalman Filter. Master's Thesis, Naval Postgraduate School Monterey, Monterey, CA, USA, 1991.
34. Saboury, J. Multi-Tap Extended Kalman Filter for a Periodic Waveform with Uncertain Frequency and Waveform Shape, and Data Dropouts. Master's Thesis, University of Wisconsin-Milwaukee, Milwaukee, WI, USA, 2019.
35. Brown, R.G.; Hwang, P.Y. *Introduction to Random Signals and Applied Kalman Filtering*, 2nd ed.; John Wiley & Sons: Toronto, ON, Canada, 1992.
36. Grewal, M.S.; Weill, L.R.; Andrews, A.P. *Global Positioning Systems, Inertial Navigation, and Integration*, 1st ed.; John Wiley & Sons: Toronto, ON, Canada, 2001.
37. Grewal, M.S.; Andrews, A.P. *Kalman Filtering Theory and Pracice Using MATLAB*, 3rd ed.; John Wiley & Sons: Toronto, ON, Canada, 2001.
38. Hofmann-Wellenhof, B.; Lichtenegger, H.; Wasle, E. *GNSS-Global Navigation Satellite Systems*; Springer: Vienna, Austria, 2008; pp. 277–307.

39. Topcon Positioning Systems, "HiPer SR Specifications", October 2021. Available online: https://www.topconpositioning.com/gnss/gnss-receivers/hiper-sr#panel-product-specifications (accessed on 21 June 2021).
40. U-Blox. "ZED-F9P-02B, U-Blox F9 High Precision GNSS Module, Data Sheet. R04". U-Blox. 2024. Available online: https://content.u-blox.com/sites/default/files/documents/ZED-F9P-02B_DataSheet_UBX-21023276.pdf (accessed on 1 August 2024).

sensors

MDPI

Article

Skeleton Reconstruction Using Generative Adversarial Networks for Human Activity Recognition Under Occlusion

Ioannis Vernikos and Evaggelos Spyrou *

Department of Informatics and Telecommunications, University of Thessaly, 35100 Lamia, Greece; ivernikos@uth.gr
* Correspondence: espyrou@uth.gr

Abstract: Recognizing human activities from motion data is a complex task in computer vision, involving the recognition of human behaviors from sequences of 3D motion data. These activities encompass successive body part movements, interactions with objects, or group dynamics. Camera-based recognition methods are cost-effective and perform well under controlled conditions but face challenges in real-world scenarios due to factors such as viewpoint changes, illumination variations, and occlusion. The latter is the most significant challenge in real-world recognition; partial occlusion impacts recognition accuracy to varying degrees depending on the activity and the occluded body parts while complete occlusion can render activity recognition impossible. In this paper, we propose a novel approach for human activity recognition in the presence of partial occlusion, which may be applied in cases wherein up to two body parts are occluded. The proposed approach works under the assumptions that (a) human motion is modeled using a set of 3D skeletal joints, and (b) the same body parts remain occluded throughout the whole activity. Contrary to previous research, in this work, we address this problem using a Generative Adversarial Network (GAN). Specifically, we train a Convolutional Recurrent Neural Network (CRNN), whose goal is to serve as the generator of the GAN. Its aim is to complete the missing parts of the skeleton due to occlusion. Specifically, the input to this CRNN consists of raw 3D skeleton joint positions, upon the removal of joints corresponding to occluded parts. The output of the CRNN is a reconstructed skeleton. For the discriminator of the GAN, we use a simple long short-term memory (LSTM) network. We evaluate the proposed approach using publicly available datasets in a series of occlusion scenarios. We demonstrate that in all scenarios, the occlusion of certain body parts causes a significant decline in performance, although in some cases, the reconstruction process leads to almost perfect recognition. Nonetheless, in almost every circumstance, the herein proposed approach exhibits superior performance compared to previous works, which varies between 2.2% and 37.5%, depending on the dataset used and the occlusion case.

Keywords: human activity recognition; occlusion; reconstruction of skeleton joints; generative adversarial networks; convolutional neural networks

Academic Editors: Manuel Gil-Martín, Rubén San-Segundo and Fernando Fernández-Martínez

Received: 31 October 2024
Revised: 28 February 2025
Accepted: 2 March 2025
Published: 4 March 2025

Citation: Vernikos, I.; Spyrou, E. Skeleton Reconstruction Using Generative Adversarial Networks for Human Activity Recognition Under Occlusion. *Sensors* **2025**, *25*, 1567. https://doi.org/10.3390/s25051567

1. Introduction

Without a doubt, the task of identifying human activities from motion data stands as one of the most challenging tasks in the broad research field of computer vision. This task can be described as the process of recognizing human behavior within a sequence of images or videos, using visual data, obtained from the motion of the human subjects in the 3D space. These behaviors include a series of successive body part movements, commonly known as "actions" or activities. Specifically, an action may be defined as a unique type of motion carried out by a human, usually of short duration [1]. Actions are not instantaneous and often involve several body parts. This category also comprises interactions, either between humans and objects or among multiple individuals involved in group activities. It should be noted that in the context of this paper, (a) the term "activity" would be used to collectively refer to all the aforementioned categories; (b) gestures, which are short and typically involve only a few body parts will not be considered; and (c) experiments will be based on segmented sequences, each containing exactly a single action to be recognized.

Camera-based Human Activity Recognition (HAR) methods are generally cost-effective since they could rely on low-cost off-the-shelf hardware, while they exhibit remarkable performance under laboratory settings. However, their performance tends to degrade in real-world scenarios due to three primary factors, i.e., changes in viewpoint/illumination, and occlusion. Variations in viewpoint may occur, for instance, when the subject is observed from different angles than the one(s) used during training. To address this, in previous work [2], it has been experimentally proven that the accuracy drop due to viewpoint changes may be mitigated by employing multiple cameras. Moreover, changes in illumination, and especially those leading to low-light conditions, predominantly affect video-based methods. Nevertheless, recent technological advances have led to the development of camera sensors capable of capturing depth information (which is unaffected by changes in illumination), thereby significantly improving performance under low-light conditions. In such cases, combining video and depth data can facilitate robust extraction of human silhouettes, for instance, as a 3D point set [3].

Therefore, occlusion remains the most significant challenge among the three issues discussed previously. In real-world conditions, it is typically caused by furniture [4], other people present [5], or even by the same subject (i.e., self-occlusion) [6], while complete occlusion of some body parts renders activity recognition impossible. Partial occlusion, however, can still affect the accuracy of recognition to some point, which varies, depending on the activity and the body part(s) affected [7]. In previous work [8], this problem has been addressed by treating it as a regression task solved using a deep neural network. This network was trained so as to reconstruct the missing skeleton joints, therefore providing a "complete" skeleton. Specifically, we used the 3D positions of skeleton joints upon removing those that were assumed to "be occluded", per case. The skeleton that occurred upon reconstruction was used to classify it into one of the predefined activities. To the best of our knowledge, this work was the first to treat the occlusion of moving body parts (i.e., subsets of skeleton joints) as a regression task. It has been demonstrated that the regression of missing joints could significantly improve the accuracy of classification. Particularly, it has been shown that a significant improvement of classification performance may be achieved by using the artificially reconstructed skeleton samples, rather than those affected by occlusion. This has been observed for any occlusion case and for almost any activity.

The novelty of this work is the reconstruction of the occluded data by treating this problem as a Generative Adversarial Network (GAN) task. Specifically, a GAN methodology is employed and various cases of partial occlusion are examined, i.e., occlusion of (a) an arm; (b) a leg; (c) both arms; (d) both legs; and (e) an arm and a leg on the same side. A Convolutional Recurrent Neural Network (CRNN) is trained to serve as the generator of the GAN, with the aim of producing a full skeleton by completing the missing parts. The input to this CRNN consists of raw 3D skeleton joint positions, but joints corresponding to the occluded parts have been removed. The output of the CRNN is a reconstructed skeleton, which is then fed into a long short-term memory (LSTM) network for classification into one of the predefined activities. For the discriminator, a simple long short-term memory (LSTM) network is used. Note that a separate network per occlusion case is trained and the effectiveness of the presented approach is evaluated by using four publicly available human activity recognition datasets and compared to a rigorous evaluation protocol, proposed in previous work [8]. As far as we know, the presented study is the first to address the occlusion of moving body parts (i.e., specific subsets of skeleton joints) within the framework of a GAN-based approach.

The rest of this paper is organized as follows: In Section 2, research works that deal with the effect of occlusion in HAR-related scenarios are presented. Then, in Section 3, the proposed reconstruction methodology is presented. Experimental results are presented in Section 3.5 and discussed in Section 5, wherein plans for future work are also presented. Finally, conclusions are drawn in Section 6.

2. Related Work

As previously noted, Human Activity Recognition (HAR) is among the most challenging research areas in computer vision, leading to a significant amount of research in this field in recent years [1]. Most of these works are based on 2D representations of skeletal motion [9–14]. Although it is both widely acknowledged and intuitive that occlusion significantly compromises the performance of HAR approaches [15], there exist only a few studies that specifically examine its impact on HAR performance or propose methods to mitigate it. In previous work [7], we have studied the effect of occlusion in 3D skeleton data. Specifically, it was simulated by removing body parts during the whole duration of the activity, and experimentally it was proved that the removal of one or both arms led to a significant drop of performance, in case of activities that are typically expressed by arm motion.

Several research works have dealt with occlusion. Specifically, Angelini et al. [16] generated synthetic partially occluded skeletons upon manually removing specific joints, per case, and showed that an increase to the recognition performance could be achieved by incorporating such occluded samples within the training procedure. Similarly, Iosifidis et al. [15] utilized a multi-camera configuration, encircling the subject, and simulated occlusion using the assumption that not all cameras can concurrently record skeleton motion. Moreover, recognition was based on fusion using solely cameras that remained "unaffected" by any case of occlusion. Moreover, Li et al. [17] used a bag of 3D points to depict poses and simulated occlusion by rejecting 3D points located in areas deemed to be affected by occlusion. Kim et al. [18] addressed the challenge of skeleton-based action recognition in occlusion scenarios by leveraging an occluded part detector and optimal joint group selection based on skeletal symmetry and angular information. In a similar manner, Chen et al. [19] introduced a Part-aware and Dual-inhibition Graph Convolutional Network (PDGCN) alongside a novel Dual Inhibition Training strategy aimed at enhancing skeleton-based action recognition in conditions of occlusion and noise. This is achieved by incorporating modules designed for occlusion simulation, global and local representation learning, and the gradual fusion of features. Gu et al. [20] generated occlusion masks, which were then utilized in both training and evaluation. In that case, a regression network was employed to reconstruct the missing skeleton parts. A single-pose embedding network was pre-trained in the work of Yang et al. [21]. The network's goal was to learn occlusion-robust representations of pose sequences. The authors used an encoder to create the pose embeddings, and also a contrastive module to render this space occlusion invariant. Ma et al [22] also worked on the problem of human pose transfer by using a Flow-based Dual Attention (FDA) GAN, addressing occlusion and deformation in feature fusion for pose generation. Their model incorporated deformable local attention and flow similarity attention, so as to handle spatial correlations. Hernandez Ruiz et al. [23] proposed a bidirectional GAN framework, namely "GAN-poser", for human motion prediction, able to handle occlusion, by learning to fill missing body parts with "plausible" motion patterns. Specifically, they enhanced the traditional GAN framework by including both forward and backward pathways, allowing better modeling of sequence dependencies and improving prediction quality. Moreover, Lee et al [24] proposed a Denoising Graph Autoencoder (DGAE) model to address the issue of missing joints in skeleton-based human action recognition (HAR) caused by occlusion and invisibility. By reconstructing missing joint coordinates with minimal error, the proposed model enhances the performance of HAR through the use of a masking Laplacian matrix to adjust feature weights and a Laplacian matrix in the decoder for reconstruction.

On the other hand, GANs have been widely used in several sensor-based HAR tasks. Irsch et al. [25] analyzed unlabeled data from wearable sensors to address data scarcity. Similarly, Li et al. [26] used GANs for sensor data augmentation. Abedin et al. [27] explored adversarial knowledge transfer for sequential sensor data. Rahman et al. [28] addressed kinematic inconsistencies in synthetic data using physics-aware models to enhance realism. Soleimani et al. [29] used GANs for cross-subject transfer learning and Jimale et al. [30] used GANs to produce synthetic data that are able to more accurately represent real data. Further, Qu et al. [31] compared the effects of GAN-generated synthetic spectrograms on classification accuracy. Wang et al. [32] explored a generative adversarial framework for sensor data generation, and Gammulle et al. [33] discussed a multi-level sequence GAN model for recognizing group activities by learning intermediate representations. Erol et al. [34] explored the use of GANs for augmenting radar-based data, improving the accuracy and robustness of HAR systems. Zadeh et al. [35] employed GANs to generate additional training data, while Wang et al. [36] leveraged GANs to combine information from multiple sensor data streams. Hasan et al. [37] introduced an innovative framework for gait recognition that tackles occlusion issues. This approach integrates an Occlusion Detection and Reconstruction (ODR) component based on 3D generative adversarial networks, along with a Feature Extraction for Gait Recognition (FEGR) module that employs both 3D and 2D CNNs to capture partwise and complete body features. Xu et al. [38] developed a technique named ActFormer, which is a GAN-based Transformer architecture designed for generating 3D human motion conditioned on actions, capable of managing both individual and multi-person interactive actions.

In the case of activity recognition using visual motion data, GANs are typically applied so as to generate realistic synthetic data to augment training datasets and/or enhance motion data quality, thereby improving the robustness and accuracy of activity recognition models in real-life scenarios. Various studies that are based on the use of 3D skeletal motion data have demonstrated the significant impact of GANs on HAR through data augmentation, transfer learning, and synthetic data generation. Fukushi et al. [39] proposed a few-shot generative model for skeleton-based human actions that leverages large source datasets to augment limited target domain samples using cross-domain and entropy regularization losses. Song et al. [40] addressed the challenge of recognizing human actions from noisy skeleton data by proposing a noise adaptation scheme based on GANs to handle noisy skeletons. Specifically, their model adapts features from noisy skeletons to a low-noise space using adversarial learning. Zalapeda et al. [41] proposed the use of synthetic data to augment the training dataset. They employed a GAN architecture to generate synthetic skeleton data and investigated the effect of different proportions of synthetic data in the training set. Degardin et al. [42] introduced Kinetic-GAN, i.e., an architecture combining GANs and Graph Convolutional Networks (GCNs) to generate realistic human body kinetics while maintaining spatial and temporal coherence. Avola et al. [43] proposed an approach for data augmentation, wherein Continuous Recurrent Neural Networks with GANs (C-RNN-GANs) were used to generate suitable synthetic action sequences to ensure robustness in cases such as incorrect skeleton extraction, perspective changes, and partial body occlusions. Shen et al. [44] addressed the challenge of limited training data in skeleton-based recognition and proposed an architecture which automatically approximates the distribution of input data and generates synthetic data, namely Imaginative GAN (Im-GAN), able to generate realistic and diverse synthetic data. Moreover, Pan et al. [45] presented a view normalization-based recognition framework to address viewpoint variance. They proposed an architecture, namely VN-GAN, wherein the generator estimates transformation parameters, applying them to normalize the viewpoint of input skeleton sequences. Liu et al. [46] proposed a generative network to generate high-quality human action data using motion style transfer and active learning. Finally, Li et al. [47] used manually defined occlusion areas, transforming a skeleton into a feature matrix. An attention model was integrated into a GAN to complete missing data of the aforementioned feature matrix.

Despite significant progress in HAR, several limitations persist, particularly regarding the challenge of occlusion. While the impact of occlusion on HAR performance is well-known, only a few studies have specifically addressed this issue, often relying on simplified scenarios where occlusion is simulated by manually removing specific joints or body parts. Such approaches may not fully capture the complexity of real-world occlusions. Additionally, methods like those proposed by Iosifidis et al. [15] depend on multi-camera configurations, which, while effective, are impractical for many real-world applications due to their cost and setup requirements. Moreover, many models are evaluated on controlled datasets with predefined occlusion conditions, limiting their generalizability to diverse or dynamic real-world environments. Reconstruction-based methods, such as those by Gu et al. [20] and Hernandez Ruiz et al. [23], have shown promise, but reconstructed joints often fail to accurately replicate real motion patterns, potentially affecting recognition performance. Advanced architectures like FDA-GAN and Denoising Graph Autoencoders (DGAEs) address occlusion with high computational demands, which can hinder real-time deployment. Furthermore, reliance on synthetic data or simulations, as seen in works like those by Yang et al. [21] and Chen et al. [19], raises concerns about their applicability in real-world scenarios. Addressing these limitations, the proposed approach focuses on developing a more robust, efficient, and generalizable approach to tackle occlusion in HAR.

3. Materials and Methods

3.1. Skeletal Data and Occlusion

3.1.1. Occlusion of Skeletal Data

Similarly to previous research efforts [2,48–53], the herein presented approach is also based on 3D trajectories of human skeletons that have been captured using the Kinect v1/v2 camera, comprising 20 or 25 joints, respectively. In Figure 1 a skeleton extracted using Kinect v1 and v2, with joints grouped to form distinct body parts, is presented.

As already mentioned in Section 1, partial occlusion may compromise the performance of HAR in real-life scenarios. Within the context of several applications, such as ambient assisted environments, AR environments, etc., occlusion typically occurs due to, e.g., activities taking place behind furniture, or due to the presence of more than one person in the same room. Of course, it should be obvious that the effects of occlusion vary depending on the activity performed. For example, the occlusion of both legs when the subject performs the action “kicking” results in a significant loss of visual information, which in turn may result to failure of recognition, while the occlusion of both arms is not expected to compromise recognition for this activity. Although the aforementioned example is quite extreme, it is common sense that partial occlusion may hinder the overall effectiveness of HAR approaches.

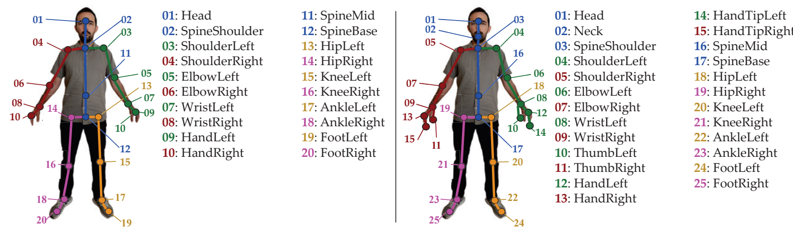


Figure 1. A human body pose with the 20 and 25 skeletal joints that are extracted using the Microsoft Kinect v1 (**left**) and v2 (**right**) cameras. Joints have been divided into subsets, each corresponding to one of the five main body parts, i.e., torso (blue), left hand (green), right hand (red), left leg (orange), and right leg (magenta). For illustrative purposes and also to facilitate comparisons between the two different versions, body parts have been colored using the same colors. Numbering follows the Kinect SDK in both cases; therefore, there exist several differences between the two versions.

3.1.2. Datasets

We are not aware of any publicly available datasets that contain real 3D occluded actions. To address this and in order to evaluate the proposed methodology, structured subsets of skeletal joints forming body parts (e.g., arms and legs) have been manually excluded from four publicly available datasets that provide 3D skeletal information. Specifically, these datasets are summarized in Table 1 and may be briefly described as follows:

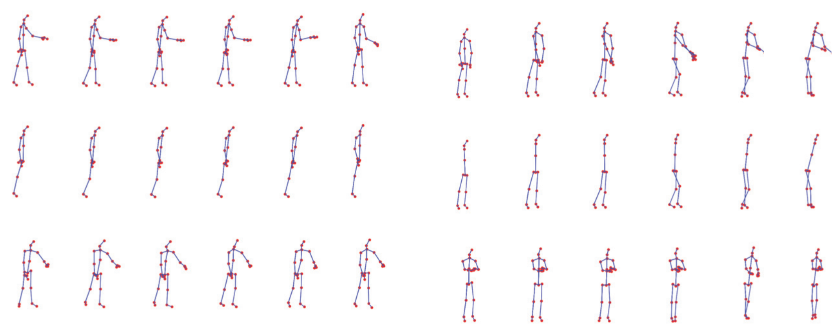
- The PKU-MMD dataset [54] is a publicly available and open-source benchmark for 3D human motion-based activity recognition. From this dataset, we opted for 11 actions that are tightly related to activities of daily living (ADLs) [51,55], i.e., eating, falling, handshaking, hugging, making a phone call, playing with a phone or tablet, reading, sitting down, standing up, typing on a keyboard, and wearing a jacket, which correspond to 21,456 data samples.
- The NTU-RGB+D dataset [56] is also a large-scale benchmark for 3D human activity analysis. From this dataset, we opted for a subset consisting of medical conditions, which includes 12 classes and 11,400 samples, i.e., sneezing/coughing, staggering, falling, headache, chest pain, back pain, neck pain, nausea/vomiting, fanning oneself, yawning, stretching, and blowing one's nose.
- SYSU 3D Human–Object Interaction (HOI) [57] is a dataset that focuses on 3D human motion-based interactions between people and objects. It contains 480 activity samples from 12 different activities, i.e., drinking, pouring, calling a phone, playing with a phone, wearing backpacks, packing backpacks, sitting on a chair, moving a chair, taking out a wallet, taking from a wallet, mopping, and sweeping. Within the aforementioned activities, 40 subjects and one of the following objects per case were involved: phone, chair, bag, wallet, mop, and besom. Each activity has 40 samples.
- The UTKinect-Action3D dataset [58] includes 10 different activities that were performed by 10 different subjects, i.e., walking, sitting down, standing up, picking up, carrying, throwing, pushing, pulling, waving hands, and clapping hands. Each activity was performed twice by each subject, resulting in a total of 200 activity instances.

From the aforementioned datasets, we only used 3D skeleton motion data and disregarded other modalities. Also, PKU-MMD and NTU-RGB were recorded using Microsoft Kinect v2 under three camera viewpoints, while SYSU-3D-HOI and UTKinect-Action3D were recorded using Microsoft Kinect v1 under a single camera viewpoint.

Table 1. Summary of datasets for human activity recognition (HAR) that have been used for the experimental evaluation of this work.

Name	Activities	Participants	Examples	Types of Activities
PKU-MMD [54]	51	66	∼20,000	Daily, sports, and health-related activities
NTU RGB+D [56]	60	40	∼56,000	Daily, interactive, and health-related actions
SYSU-3D-HOI [57]	12	40	480	Human–Object Interactions
UTKinect-Action-3D [58]	10	10	200	Interactive and gesture-based actions

It should be noted that most public motion-based datasets, such as the ones herein used for evaluation purposes, have been created under ideal laboratory conditions, and thus occlusion is prevented. Since the creation of a large-scale dataset is a time-consuming task, it was decided to follow an approach such as the one of [7,8,20], i.e., subsets of joints that correspond to body parts will be manually discarded, assuming that these parts remain occluded during the whole duration of each activity (see Figure 1). Moreover, and for the sake of explanation, two visual examples of activities with/without occlusion are illustrated in Figure 2, where the loss of visual information is easily comprehensible. Specifically, an example of a successfully reconstructed skeleton, which leads to correct classification, and another example of an unsuccessfully reconstructed skeleton, which leads to incorrect classification, are both therein illustrated.



(**a**) (**b**)

Figure 2. Example skeleton sequences of the activities (**a**) *handshaking* and (**b**) *hugging other person* from the PKU-MMD dataset, captured by Microsoft Kinect v2. First row: original skeletons, including all 25 joints (i.e., without any occlusion); second row: joints corresponding to (**a**) left arm; (**b**) both arms (see Figure 1) have been discarded (i.e., the skeleton is partially occluded); third row: skeletons have been reconstructed using the proposed deep regression approach. The example of (**a**) is successfully reconstructed and correctly classified, while the example of (**b**) is unsuccessfully reconstructed and incorrectly classified.

3.2. Generative Adversarial Networks

A generative adversarial network (GAN) [59] is a framework for machine learning, mainly used in generative AI [60]. In brief, it consists of two neural networks that take part in a game. Within this game, the gain of one network leads to a loss of the other. The one network is the generator, while the other is the discriminator. The GAN aims to learn how to generate "new" examples which follow the same statistics as those that have been used for training. For example, in the presented approach, a GAN trained on skeleton sequences shall be used; the goal of its generator will be to generate new skeleton sequences that will be “realistic” compared to the real ones that have been used for training. This is achieved based on training through feedback by the discriminator, which is able to discriminate between “realistic” and “non-realistic” images. Both generator and discriminator are dynamically updated, i.e., they are trained in a way that the the former learns to “fool” the latter. Specifically, a given sample that is generated by the generator should be mapped to a specific distribution; each of them should be recognized as belonging to the true distribution or not by the discriminator.

3.2.1. Generator

The generator is formulated as a *regression* task, i.e., the one presented in [8]. Specifically, let $\mathbf{X}$ denote the original skeleton sequence and $\mathbf{X_o}$ the sequence resulting from occlusion. The aim of regression, i.e., of the generator, is to estimate a set of parameters β of a given function f, so that $\mathbf{X} = f(\mathbf{X_o}, \beta) + \epsilon$, where ϵ is some error value, can be minimized. The Convolutional Recurrent Neural Network (CRNN) model of previous work [8] is employed as the generator model for the GAN framework to achieve the objective. Therefore, the purpose of this model is to execute f and learn β (i.e., its weights) in order to reduce ϵ. When given an occluded skeleton sequence $\mathbf{X_o}$, the network generates a reconstructed skeleton sequence $\mathbf{X}_r$, which is an approximation of $\mathbf{X}$, by filling in the missing (occluded) data (joints). Also, in this case, the CRNN includes an LSTM whose goal is to capture temporal information of skeletal data. Figures 3 and 4 illustrate the architectures of the CRNN and the LSTM networks, respectively.

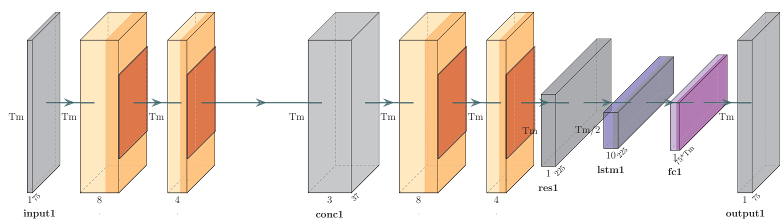


Figure 3. The architecture of the generator of the proposed GAN.

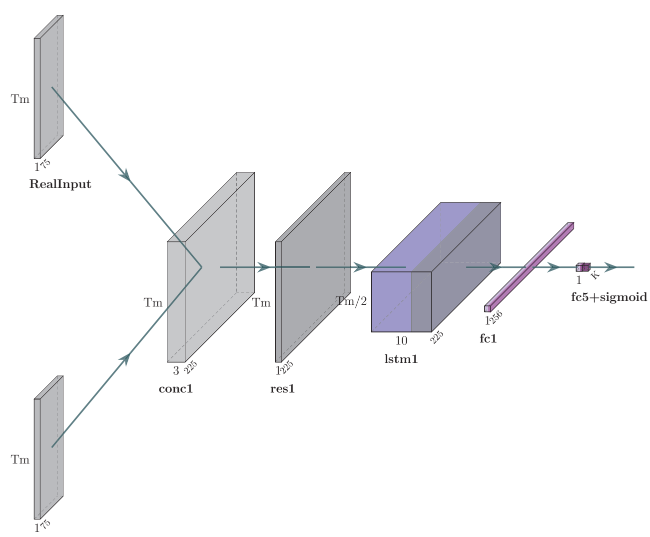


Figure 4. The architecture of the discriminator of the proposed GAN architecture.

3.2.2. Discriminator

The discriminator uses as input a generated image and a real image and reshapes them according to the timeframes that have been set. Its architecture is quite simple: it first feeds the aforementioned images into an LSTM layer, followed by two dense layers, the first comprising 256 neurons and the last solely 1 neuron that produces a binary result, i.e., to determine whether the image is real or not. The discriminator loss is a sigmoid cross-entropy loss of the real and generated images. The architecture of the discriminator is illustrated in Figure 4.

3.3. Classification

An occluded activity sample (i.e., a sample with missing body parts resulting from occlusion) is fed as input to the trained CRNN network, whose role is to reconstruct the missing skeletal data. For the sake of explanation, two visual examples of an activity before and after reconstruction are shown in Figure 2. Upon reconstruction, the proposed approach proceeds with the classification of this sample into one of the predefined activities. For classification, a LSTM network with 1 layer is used. It should be herein emphasized that the CRNN network is trained using only *full skeletal data* $\mathbf{X}$, i.e., not affected by occlusion of any body part(s). Conversely, during evaluation, the input to the LSTM network consists of the corresponding skeletal sequences $\mathbf{X}_r$, which have been reconstructed from $\mathbf{X}$. A visual overview of the proposed approach is illustrated in Figure 5.

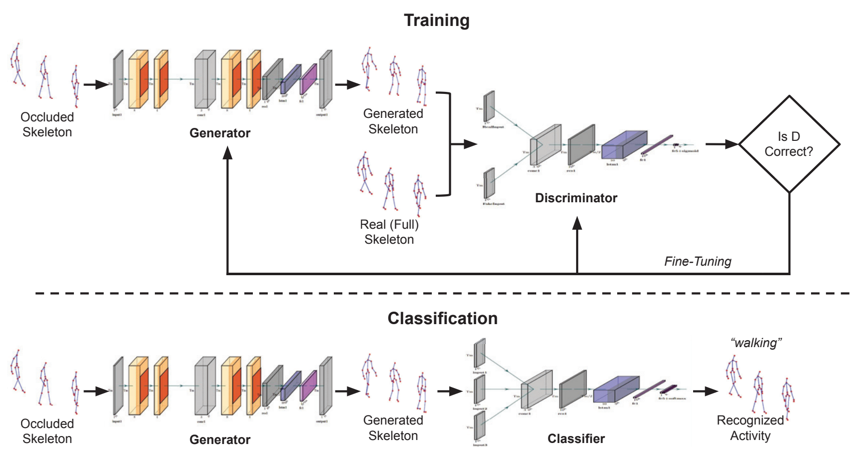


Figure 5. A visual overview of the proposed approach.

To assess the performance of the herein proposed skeleton reconstruction approach, a classifier to measure the accuracy of the model is used. The architecture of the classifier is illustrated in Figure 6. Specifically, in case wherein three different camera viewpoints are used, the input from all cameras are combined into one, reshaped to fit the predetermined timeframes, and then fed into an LSTM layer. Following this step, two dense layers are used, the first comprising 256 neurons and the last comprising 1 neuron that produces a result which is an indication of the user's activity (see Figure 6). For datasets with a single camera viewpoint, the same network is used, hence with only one input (see Figure 7).

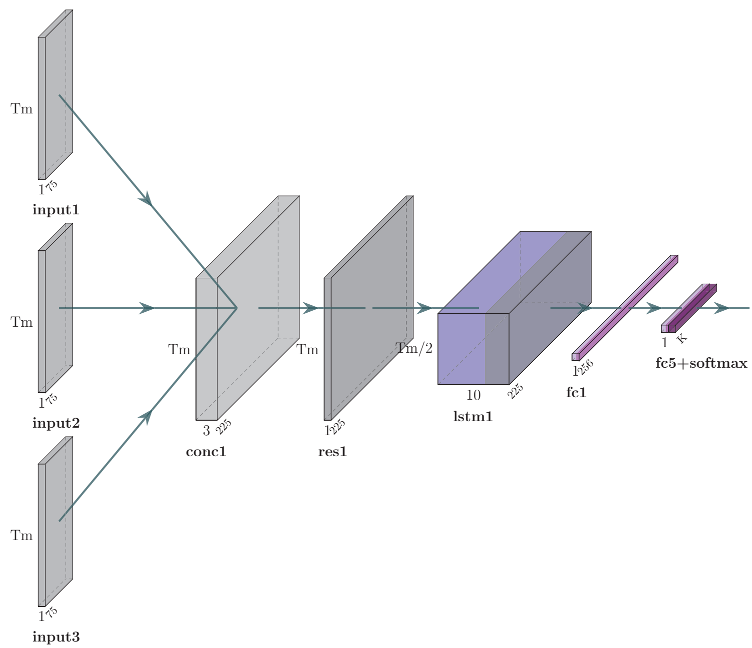


Figure 6. The architecture of the classifier of the proposed approach for the three-camera case.

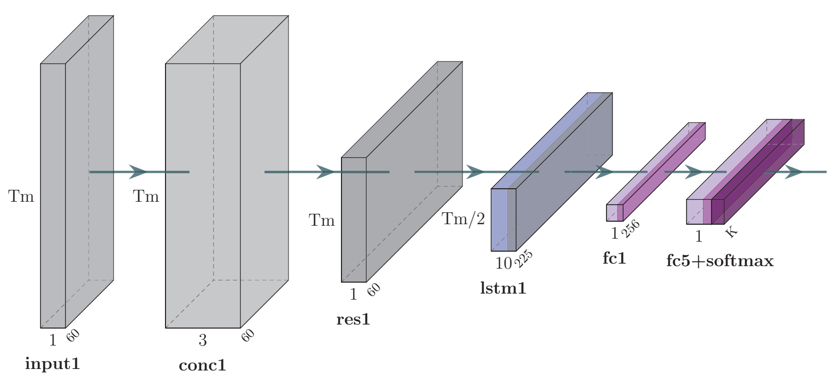


Figure 7. The architecture of the classifier of the proposed approach for the one-camera case.

3.4. The GAN Objective

In this work, the objective of the Pix2Pix GAN framework [61] is adopted. Specifically, the objective L_{cGAN} of a conditional Generative Adversarial Network (GAN) is described by the following:

$$L_{\text{cGAN}}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z)))] \ . \tag{1}$$

Previous approaches have suggested combining the GAN objective with a more traditional loss, such as the L1 distance; in that case, it should be "close" to the real output:

$$L_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x, z)\|_1] \tag{2}$$

Thus, upon combination, the final objective is

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \tag{3}$$

3.5. Experiments

3.5.1. Experimental Setup and Network Training

Experiments were conducted on a personal workstation with an Intel™i7 4770 4-core processor running at 3.40 GHz and 32GB RAM, using an NVIDIA™Geforce RTX 3070 GPU with 8 GB VRAM and Ubuntu 20.04 (64 bit). The deep architecture was implemented in Python using Keras 2.4.3 [62] with the Tensorflow 2.5 [63] backend. All data pre-processing and processing steps were implemented in Python 3.9 using NumPy and SciPy. For the training of the generator, the LeakyReLU activation function was used, except for the LSTM layer where the tanh function was used, and the last dense layer where the linear activation function was used. For the training of the classifier, the LeakyReLU and tanh activation functions were used, respectively, except for the last layer, where the sigmoid activation function was used. The batch size was set to 5 and 10 for the training of the classifier and the GAN, respectively. The Adam optimizer was utilized in both cases, the dropout was configured to 0.25, the learning rate to 0.001, and the training was conducted for 200 epochs, using the loss of the validation set calculated via MSE as an early stopping method to prevent overfitting. In all cases, 80% of the available activity samples was used for training, 10% for validation, and the remaining 10% for evaluation.

As in previous work [8], the herein presented approach takes temporal sequences of 3D skeleton data as input, while interpolation is applied to set the duration of all activity samples equal to T_m, which is the size of the sample of max duration. Note that experiments consider both single- and multi-view datasets. In the latter case, all available views are utilized by feeding the network with all three skeleton sequences. It is also assumed that occlusion affects the same body part(s), in every view, and within the whole duration of the activity. The presented approach is based on the idea that since occlusion leads to missing body parts (i.e., in this case some of the skeleton joints have been discarded, so the respective coordinates are missing), the problem of "reconstructing" the respective values of missing body parts could be devised as a generative adversarial network (GAN) task. Again, as in [8], one network per occlusion case is trained, resulting in eight different networks. Thus, by giving a sequence of skeletal data as input, while some action is being performed, the missing skeletal joints are initially identified and then fed to the appropriate trained network, based on the corresponding occlusion case. Then, using this network, it is classified into an activity class. Note that even though the proposed approach requires more memory to store all the aforementioned networks, this is compensated by the significant increase in performance compared to the use of a single network. It should be emphasized that all the classifiers are trained exclusively using samples not affected by occlusion.

At this stage, a sample with missing skeletal joints due to occlusion can be used as input into the trained generator, which then reconstructs the missing skeletal data. An example of this activity before and after reconstruction is shown in Figure 2. After reconstruction, the data can be classified into one of the pre-defined classes using the classifier. During the testing phase, the input of the classifier is a reconstructed skeletal sequence $\mathbf{X}_r$.

3.5.2. Evaluation Protocol

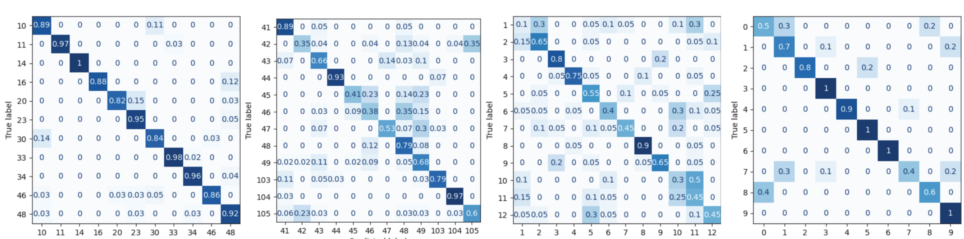
To experimentally evaluate the proposed methodology, the following series of experiments is considered:

i. Removal of structured sets of skeletal joints, corresponding to body parts, to simulate occlusion (see Figure 1). Specifically, as already mentioned, cases of part removal include (a) left arm; (b) right arm; (c) both arms; (d) left leg; (e) right leg; (f) both legs; (g) left arm and left leg; (h) right arm and right leg. We used an LSTM network that had been trained using exclusively samples that were not affected by occlusion, and also the skeletons were reconstructed using a GAN;
ii. A "baseline" approach, where both training and evaluation of the LSTM took place using exclusively samples not affected by occlusion;
iii. A "reference" approach where training of the LSTM took place using exclusively samples not affected by occlusion, but was evaluated using occluded samples;
iv. An approach wherein samples affected by occlusion were included in the training process of the LSTM, while validation was performed exclusively using occluded samples. Here, a subset equal to 10% of the non-occluded samples of the training set was selected. From these samples, all eight cases of occlusion have been generated, thus "augmenting" the initial training data by 80%. Note that a single network was used for all eight cases of occlusion.

Note that in this study, we strictly adhere to the predefined evaluation protocols associated with each dataset, which dictate a single training and testing procedure on fixed splits, so as to ensure comparability of results.

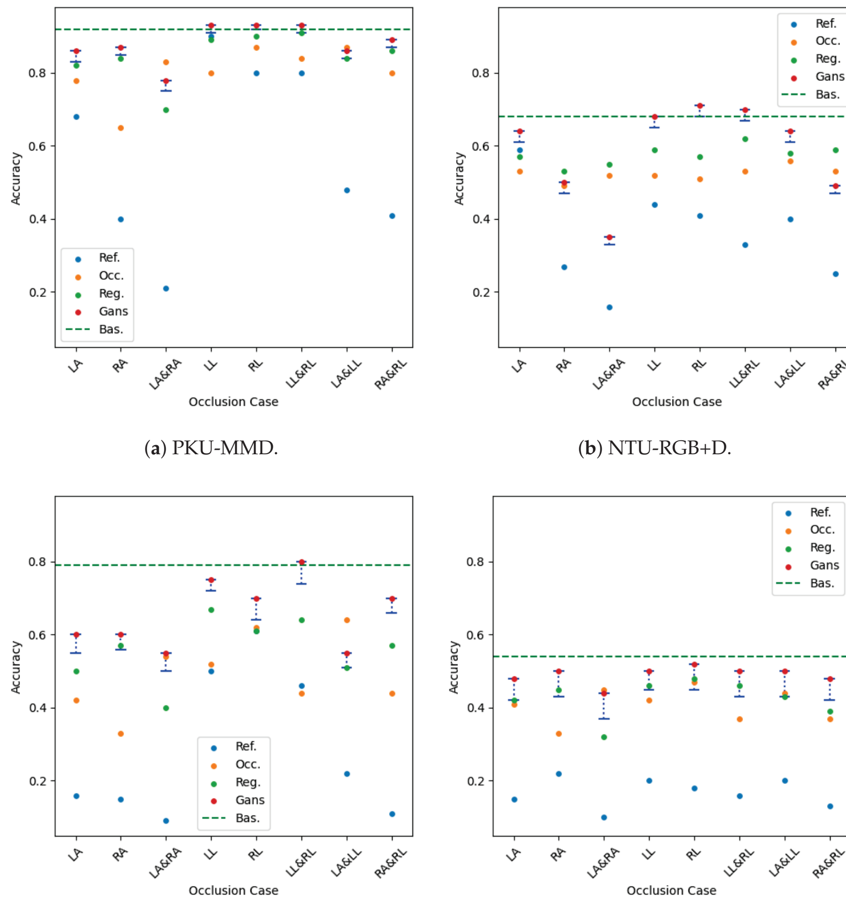
4. Results

Experimental results for all datasets are depicted in Tables 2–5, while most extensive results, per class and per occlusion case, are depicted in the Appendix and specifically in Tables A1–4. The following metrics have been extracted: accuracy per class, and F_1-score per class and weighted accuracy, where class weights were calculated based on the class distribution. Moreover, confusion matrices for all datasets in the case of the baseline experiment are depicted in Figure 8. Note that in Tables 2–5 and in Figure 9, confidence intervals of the proposed approach are presented and compared to the best results reported in previous works. Furthermore, the case of the classification of reconstructed samples is depicted in Figures 10–13. We also performed comparisons using the aforementioned protocol with two previous works. Specifically, in [64], a data augmentation approach was presented, wherein a CNN was trained with the addition of artificially occluded samples in the training set, while in [8], a deep regression approach for skeleton reconstruction was presented. Note that although the same evaluation protocol and the same datasets are used as in these works, the herein presented work formulates the problem of reconstruction as a GAN task, and is hence based on a novel GAN architecture.



(**a**) PKU-MMD. (**b**) NTU-RGB+D. (**c**) SYSU-3D-HOI. (**d**) UTKinect-Action-3D.

Figure 8. Normalized confusion matrices for classification for all datasets, without removing any body part.



(**a**) PKU-MMD. (**b**) NTU-RGB+D.

(**c**) UT-Kinect-Action-3D. (**d**) SYSU-3D-HOI.

Figure 9. Confidence intervals using the proposed approach on all datasets, compared with the best weighted accuracies reported in previous works. In case of the proposed approach, red dot denotes the upper bound of the confidence interval, i.e., the best weighted accuracy achieved.

Table 2. Results on PKU-MMD dataset. "Bas."/"Gans"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/ Regression/training with occluded samples/reference case. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers denote Weighted Accuracy, numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". In case of Gans, by "max" and "min" we denote the upper and the lower bounds of the confidence interval.

	None		LA	RA	LA+RA	LL	RL	LL+RL	LA+LL	RA+RL
Gans	-	max	**0.86**	**0.87**	0.78	**0.93**	**0.93**	**0.93**	0.86	**0.89**
		min	0.83	0.85	0.75	0.91	0.92	0.91	0.84	0.87
Reg.	-		0.82	0.84	0.70	0.89	0.90	0.91	0.84	0.86
Occ.	-		0.78	0.65	**0.83**	0.80	0.87	0.84	**0.87**	0.80
Ref.	-		0.68	0.40	0.21	0.90	0.80	0.80	0.48	0.41
Bas.	0.92	-	-	-	-	-	-	-	-	-

Table 3. Results on NTU-RGB+D dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/ Regression/training with occluded samples/reference case. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers denote Weighted Accuracy, numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". In case of Gans, by "max" and "min" we denote the upper and the lower bounds of the confidence interval.

	None		**LA**	**RA**	**LA+RA**	**LL**	**RL**	**LL+RL**	**LA+LL**	**RA+RL**
Gans	-	max	**0.64**	0.50	0.35	**0.68**	**0.71**	**0.70**	**0.64**	0.49
		min	0.61	0.47	0.33	0.65	0.68	0.67	0.61	0.47
Reg.	-		0.57	**0.53**	**0.55**	0.59	0.57	0.62	0.58	**0.59**
Occ.	-		0.53	0.49	0.52	0.52	0.51	0.53	0.56	0.53
Ref.	-		0.59	0.27	0.16	0.44	0.41	0.33	0.40	0.25
Bas.	0.68	-	-	-	-	-	-	-	-	-

Table 4. Results on UTKinect-Action3D dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/Regression/training with occluded samples/reference case. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers denote Weighted Accuracy, numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". In case of Gans, by "max" and "min" we denote the upper and the lower bounds of the confidence interval.

	None		**LA**	**RA**	**LA+RA**	**LL**	**RL**	**LL+RL**	**LA+LL**	**RA+RL**
Gans	-	max	**0.60**	**0.60**	**0.55**	**0.75**	**0.70**	**0.80**	0.55	**0.70**
		min	0.55	0.56	0.50	0.72	0.64	0.74	0.51	0.66
Reg.	-		0.50	0.57	0.40	0.67	0.61	0.64	0.51	0.57
Occ.	-		0.42	0.33	0.54	0.52	0.62	0.44	**0.64**	0.44
Ref.	-		0.16	0.15	0.09	0.50	0.61	0.46	0.22	0.11
Bas.	0.79	-	-	-	-	-	-	-	-	-

Table 5. Results on SYSU-3D-HOI dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/Regression/training with occluded samples/reference case. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers denote Weighted Accuracy, numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". In case of Gans, by "max" and "min" we denote the upper and the lower bounds of the confidence interval.

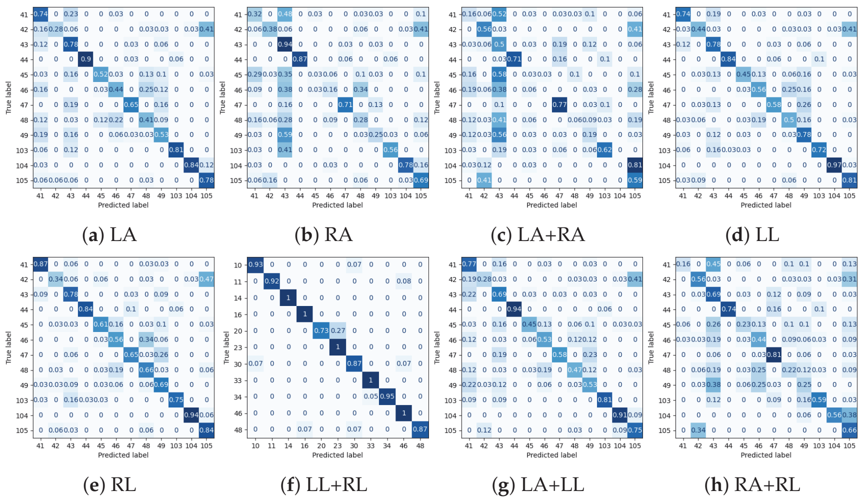
	None		**LA**	**RA**	**LA+RA**	**LL**	**RL**	**LL+RL**	**LA+LL**	**RA+RL**
Gans	-	max	**0.48**	**0.50**	0.44	**0.50**	**0.52**	**0.50**	**0.50**	**0.48**
		min	0.42	0.43	0.37	0.45	0.45	0.43	0.43	0.42
Reg.	-		0.42	0.45	0.32	0.46	0.48	0.46	0.43	0.39
Occ.	-		0.41	0.33	**0.45**	0.42	0.47	0.37	0.44	0.37
Ref.	-		0.15	0.22	0.10	0.20	0.18	0.16	0.20	0.13
Bas.	0.54	-	-	-	-	-	-	-	-	-

In the case of the PKU-MMD dataset, the weighted accuracy (WA) was 0.92 without any body part removal. Specifically, it ranged between 0.21 and 0.90 in case of some body part removal, while it ranged between 0.78 and 0.93 upon reconstruction with the GAN. In all cases, significant improvement was observed in terms of WA. Moreover, reconstruction with the GAN outperformed the regression approach [8] in all cases, and the augmentation approach [64] in 6 out of 8 cases. Since most of the activities used to evaluate our approach primarily involve upper body motion (i.e., are expressed by the motion of left and/or right arm), this is also reflected to the results of Table A1, wherein it may be observed that in cases of occluded arms, the improvement is significantly large, with the most notable example being the case of both arms, wherein WA improves from 0.21 to 0.78. Upon careful observation of the confusion matrices depicted in Figure 11, for each occlusion case, the following should be noticed when comparing the case where all joints were used:

a. In the case of any occluded arm, class *make a phone call/answer phone* is often confused with *playing with phone/tablet*. In the case of the occluded right arm, class *eat meal/snack* is very often confused with *reading*. This happens less often in the case of the occluded left arm. Also, in a few cases, class *handshaking* is confused with *hugging*, and class *reading* with *typing on a keyboard*.
b. In the case of any occluded leg, class *make a phone call/answer phone* is often confused with *playing with phone/tablet*. Also, in the case of occluded left leg, class *reading* is often confused with *eat meal/snack*, while in the case of occluded right leg, class *eat meal/snack* is very often confused with *reading*.
c. In the case of occluded left arm and left leg, class *make a phone call/answer phone* in the majority of testing examples is confused with *playing with phone/tablet* and class *hugging* with *playing with phone/tablet* or *handshaking*.
d. In the case of occluded right arm and right leg, class *make a phone call/answer phone* is often confused with *playing with phone/tablet* or *handshaking*, *eat meal/snack* is very often confused with *reading*, and class *handshaking* is often confused with *make a phone call/answer phone*.

In all cases, the majority of classes demonstrate excellent performance, equivalent to the case without any occlusion. Compared to the augmentation approach of [64], it should be noted that reconstruction using GANs was superior in 6 out of the 8 cases of occlusion. Specifically, augmentation showed superior performance in the case of both occluded arms and in the case of occluded left arm and left leg, although in that case, reconstruction using GANs exhibited equivalent performance. Finally, as shown in Figure 9a, in most cases, the lower bound of the weighted accuracy's confidence interval exceeds a better performance than all other approaches.

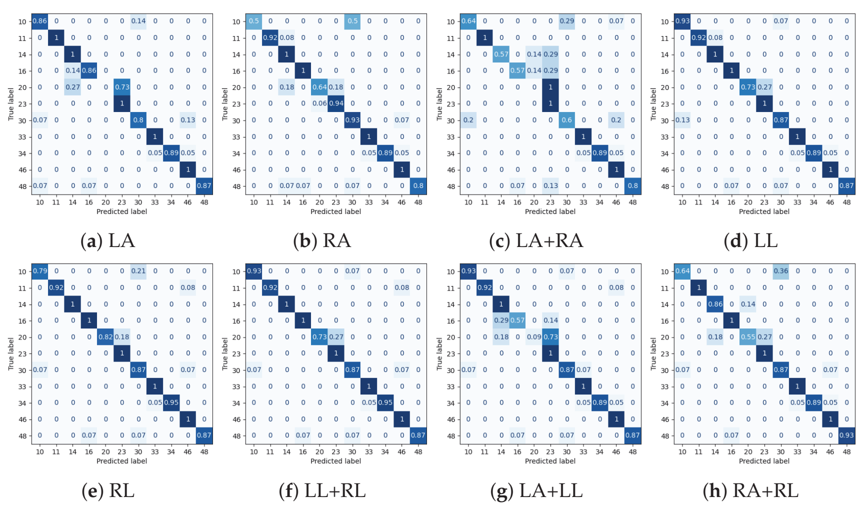
In the case of the NTU-RGB+D dataset, the WA was 0.68 without any body part removal and ranged between 0.16 and 0.59 in the case of some body part removal, while it ranged between 0.35 and 0.71 upon reconstruction with the GAN. In that case, in 5 out of 8 cases, significant improvement was observed in terms of WA, while performance was almost equal with the other two approaches in the case of removal of right arm or right arm and right leg. In the remaining three cases, the regression approach demonstrated the best performance. Moreover, reconstruction with GAN outperformed the augmentation approach in 7 out of 8 cases. Since activities used to evaluate the proposed approach mainly consisted of upper body motion, also in the case of the NTU-RGB+D dataset, in the results of Table A2, it could be observed that in all the remaining cases of occluded arms, the improvement of WA is large, with the most notable example being the case of both arms, wherein WA improves from 0.16 to 0.35, although the other two approaches demonstrated superior performance in that case. Upon careful observation of the confusion matrices depicted in Figure 10, and considering the classification results without occlusion of any part, for each occlusion case it should be noticed that most activities are affected by the occlusion of the right arm and performance is far from being excellent in the majority of these cases. Notably, reconstruction with GANs exhibits best performance in case of occluded legs. Finally, also in this dataset and as shown in Figure 9b, in most cases, the lower bound of the weighted accuracy's confidence interval exceeds best performance of all other approaches.



(**a**) LA (**b**) RA (**c**) LA+RA (**d**) LL

(**e**) RL (**f**) LL+RL (**g**) LA+LL (**h**) RA+RL

Figure 10. Normalized confusion matrices for classification for the NTU-RGB+D dataset. LA, RA, LL and RL correspond to cases of occluded Left Arm, Right Arm, Left Leg and Right Leg, respectively.



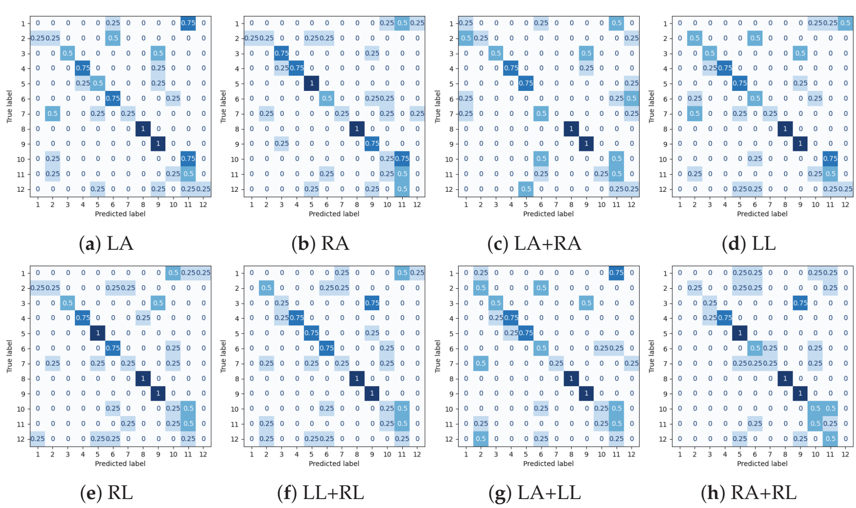
(**a**) LA (**b**) RA (**c**) LA+RA (**d**) LL

(**e**) RL (**f**) LL+RL (**g**) LA+LL (**h**) RA+RL

Figure 11. Normalized confusion matrices for classification for the PKU-MMD dataset. LA, RA, LL and RL correspond to cases of occluded Left Arm, Right Arm, Left Leg and Right Leg, respectively.

In the case of the SYSU-3D-HOI dataset, the WA was 0.54 without any body part removal and ranged between 0.10 and 0.22 in the case of some body part removal, while it ranged between 0.44 and 0.52 upon reconstruction with the GAN, as may be observed in the results of Table 4. Moreover, the reconstruction with GANs outperformed the other two methods in 7 out of 8 cases. In every case of occlusion, significant improvement was observed in terms of WA when compared to the reference case. It should be noticed that due to the small size of this dataset, performance was inadequate in several classes even without occlusion, while reconstruction in several cases exhibited superior performance. It was quite a surprise to observe that reconstruction with GANs steadily exhibited almost constant performance for all occlusion cases, whilst in many occasions it was also superior to the baseline approach, i.e., to the absence of occlusion. Also, due to the small size of the dataset, often several classes fail to be recognized. Finally, also in this dataset and as shown in Figure 9b, in several cases, the lower bound of the weighted accuracy's confidence interval does not exceed best performance of all other approaches; however, this was expected due to the small dataset size.

In the case of the UTKinect-Action-3D dataset, the WA was 0.79 without body part removal, and ranged between 0.09 and 0.61 in the case of some body part removal and also between 0.55 and 0.80 upon reconstruction with GAN. Moreover, reconstruction with GANs outperformed the other two methods in 7 out of 8 cases. Since activities used to evaluate the proposed approach mainly consisted of upper body motion, also in the case of the UTKinect-Action-3D dataset, in the results of Table 3 it could be observed that in all the remaining cases of occluded arms, the improvement of WA is exceptional, with the most notable examples the cases of occluded right arm and right leg, wherein WA improves from 0.11 to 0.70 and right arm and left arm wherein WA improves from 0.09 to 0.55. Upon careful observation of the confusion matrices depicted in Figure 13 and considering the classification results without occlusion of any part, for each occlusion case we should notice that occlusion affects half of the activities, especially those that are based on the motion of arms. Finally, also in this dataset and as shown in Figure 9c, in most cases the lower bound of the weighted accuracy's confidence interval exceeds best performance of all other approaches.

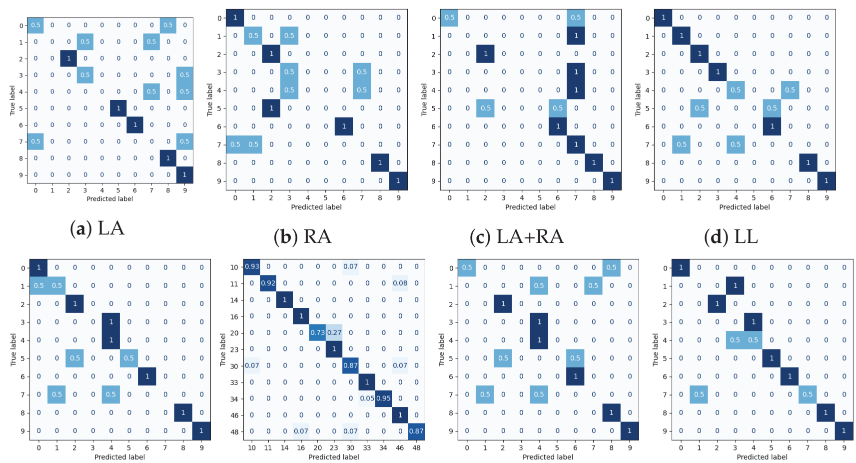


(**a**) LA (**b**) RA (**c**) LA+RA (**d**) LL

(**e**) RL (**f**) LL+RL (**g**) LA+LL (**h**) RA+RL

Figure 12. Normalized confusion matrices for classification for the SYSU-3D-HOI dataset. LA, RA, LL and RL correspond to cases of occluded Left Arm, Right Arm, Left Leg and Right Leg, respectively.

In all cases, the occlusion of some body parts leads to a severe drop in performance; in many cases, the accuracy and F_1-score of some classes dropped to zero/near zero values. In some instances, the reconstruction was successful, resulting in nearly perfect recognition. However, even in these cases, the classification of reconstructed samples for several activities and specific occlusion scenarios showed slightly inferior performance compared to the occluded samples. We believe that in these cases, the occluded body part is less "relevant" for those activities. Nonetheless, there are instances where the reconstruction approach may fail, leading to misleading joint positions, as demonstrated in the example of Figure 2.



(**a**) LA (**b**) RA (**c**) LA+RA (**d**) LL

(**e**) RL (**f**) LL+RL (**g**) LA+LL (**h**) RA+RL

Figure 13. Normalized confusion matrices for classification for the UT-Kinect-Action-3D dataset. LA, RA, LL and RL correspond to cases of occluded Left Arm, Right Arm, Left Leg and Right Leg, respectively.

5. Discussion and Future Work

The results across all datasets demonstrate that the occlusion of body parts significantly impacts recognition performance, often causing a severe drop in accuracy and F_1-scores, with some classes dropping to near-zero values. However, reconstruction using GANs consistently showed notable improvements, outperforming alternative approaches in most cases. Specifically, in datasets like PKU-MMD and UTKinect-Action-3D, the weighted accuracy improved dramatically in scenarios involving occluded arms, underscoring the effectiveness of GAN-based reconstruction in handling upper-body motion activities.

Despite these successes, certain limitations persist. For activities less dependent on the occluded body parts, reconstruction sometimes exhibited slightly inferior performance compared to the baseline, suggesting a diminished relevance of those parts for the specific activities. Additionally, while GANs excelled in mitigating the effects of occlusion, occasional reconstruction failures resulted in misleading joint positions, potentially affecting classification accuracy. These findings highlight both the potential and the challenges of leveraging GANs for robust activity recognition in occlusion scenarios.

Future research work may target various aspects of the HAR problem with a focus on occlusion. Our regression approach could be enhanced and improved by, for example, replacing the interpolation step currently used with a temporal augmentation approach similar to that of Kwon et al. [65]. Additionally, incorporating other state-of-the-art architectures, such as transformers [66], into the classification process could yield significant improvements. Moreover, the aspect of occlusion in HAR, including scenarios like temporally partial occlusion, warrants further investigation. Since handcrafted features have been experimentally shown to enhance the recognition performance of deep learning approaches [67], experimenting with other feature extraction methodologies based on the geometry and motion of skeletons, such as the approach proposed by Avola et al. [43], would be of great interest. Finally, we also believe that conducting real-life experiments in an assistive living environment would be extremely valuable.

6. Conclusions

In this paper, a methodology for reconstructing human skeleton sequences was developed by employing a generative adversarial network. The generator in this network was a convolutional recurrent neural network, which was trained to generate the complete skeleton by filling in the missing parts. The input to the generator consisted of the raw 3D positions of the skeleton joints, with the occluded joints removed. The output of the generator was a reconstructed skeleton, which was then passed through a long short-term memory network for classification into one of the predefined activities. For the discriminator, a simple long short-term memory network trained only on non-occluded samples was used. It was demonstrated that in all scenarios, the occlusion of certain body parts resulted in a significant decline in performance, although the reconstruction process achieved nearly perfect recognition in some cases. Through experimental evaluation, it has also been demonstrated that the proposed approach offers improved performance compared to previous methodologies, which varied between 2.2% and 37.5%, depending on the chosen dataset and the specific occlusion scenario.

To conclude, in real-life applications, i.e., in dynamic and real-life scenarios, human activity recognition from visual data faces several limitations when dealing with occlusion. Real-life occlusions are inherently unpredictable, caused by factors such as environmental obstacles, people entering or leaving the frame, and self-occlusion, making them difficult to accurately model in controlled studies. Furthermore, most HAR datasets are collected in constrained environments with limited variability and predefined activities, reducing their generalizability to diverse real-world conditions. Sensor limitations, such as restricted camera angles and the impact of environmental factors like lighting or background clutter, intensify the challenges of recognizing occluded actions in dynamic settings. Current models often depend on synthetic or static occlusion scenarios for training, which fail to capture the complexity of real-world occlusions, and their computational demands can hinder real-time deployment. Additionally, standard evaluation metrics may not reflect the nuances of occlusion in real-world applications, and overlapping activities can create further ambiguity in recognition. Finally, ethical and privacy concerns during data collection in public spaces limit the scope of real-world studies, while participant behavior under observation may introduce biases. Addressing these limitations requires the use of multi-sensor systems, dynamic dataset creation, advanced occlusion-handling models, and robust ethical frameworks for real-world applicability.

Author Contributions: Conceptualization I.V. and E.S.; methodology, I.V.; validation, I.V.; data curation, I.V.; writing—original draft preparation, I.V. and E.S.; writing—review and editing, E.S.; supervision, E.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The PKU-MMD dataset is available at https://www.icst.pku.edu.cn/struct/Projects/PKUMMD.html (accessed on 3 March 2025). The NTU-RGB+D dataset is available at https://rose1.ntu.edu.sg/dataset/actionRecognition/ (accessed on 3 March 2025). The UT-Kinect-3D dataset is available at https://cvrc.ece.utexas.edu/KinectDatasets/HOJ3D.html (accessed on 3 March

2025). The SYSU-3D-HOI dataset is available at https://www.isee-ai.cn/~hujianfang/ProjectJOULE.html (accessed on 3 March 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

2D	2-Dimensional
3D	3-Dimensional
AR	Augmented Reality
CRNN	Convolutional Recurrent Neural Network
CsiGAN	Channel State Information Generative Adversarial Network
ExGANs	Exemplar GANs
FDA	Flow-based Dual Attention
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
HOI	Human–Object Interaction
HAR	Human Activity Recognition
LSTM	Long Short-Term Memory
MSE	Mean Squared Error
RNN	Recurrent Neural Network
WA	Weighted Accuracy

Appendix A

Table A1. Results on PKU-MMD dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/ Regression/training with occluded samples/reference case, Acc, F_1 and WA denote Accuracy, F_1 score and Weighted Accuracy, respectively. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". Numbers in italics denote the corresponding classes, as follows: 10: eat meal/snack, 11: falling, 14: handshaking, 16: hugging other person, 20: make a phone call/answer phone, 23: playing with phone/tablet, 30: reading, 33: sitting down, 34: standing up, 46: typing on a keyboard, 48: wear jacket.

		None		**LA**				**RA**				**LA+RA**				**LL**				**RL**				**LL+RL**				**LA+LL**				**RA+RL**		
Class	**Metric**	**Bas.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**
10	Acc.	0.90	**0.86**	0.84	0.43	0.11	**0.50**	0.45	0.36	0.03	0.64	0.58	**0.64**	0.00	**0.93**	0.79	0.71	0.92	**0.79**	**0.79**	0.50	0.66	0.93	**0.87**	0.71	0.74	**0.93**	0.68	0.79	0.21	**0.64**	0.66	**0.71**	0.24
	F_1	0.86	**0.86**	0.75	0.60	0.15	**0.67**	0.57	0.32	0.05	0.69	0.55	**0.78**	0.00	**0.90**	0.86	0.83	0.84	**0.85**	0.81	0.67	0.72	0.93	**0.82**	0.74	0.78	**0.93**	0.70	0.88	0.25	0.75	0.69	**0.77**	0.36
11	Acc.	0.97	**1.00**	0.97	**1.00**	0.97	0.92	**0.97**	0.85	0.97	1.00	0.97	**1.00**	0.00	0.92	**0.97**	0.92	0.97	0.92	0.95	**1.00**	0.97	0.92	**0.97**	0.85	0.97	0.92	0.97	**1.00**	**1.00**	**1.00**	0.97	**1.00**	0.97
	F_1	0.99	**1.00**	0.99	0.96	0.90	0.96	**0.97**	0.88	0.99	1.00	0.95	**1.00**	0.00	0.96	0.91	0.92	**0.99**	0.96	0.97	**1.00**	0.99	0.96	**0.99**	0.88	0.99	0.96	**0.97**	0.96	0.70	**1.00**	0.99	**1.00**	0.92
14	Acc.	1.00	**1.00**	0.88	1.00	**1.00**	**1.00**	**1.00**	0.29	1.00	0.57	**1.00**	1.00	0.81	**1.00**	**1.00**	**1.00**	0.94	**1.00**	**1.00**	0.86	1.00	**1.00**	1.00	0.57	1.00	**1.00**	**1.00**	**1.00**	0.94	0.86	**1.00**	0.29	1.00
	F_1	1.00	0.78	0.93	0.58	**0.97**	0.78	**1.00**	0.27	1.00	0.73	**0.97**	0.67	0.90	0.93	**0.97**	0.64	0.94	**1.00**	**1.00**	0.71	0.91	**1.00**	1.00	0.62	0.91	0.78	**1.00**	0.78	0.91	0.80	**0.97**	0.33	0.86
16	Acc.	0.88	0.86	**0.94**	0.14	0.88	**1.00**	0.94	0.14	0.88	0.57	**0.88**	0.43	0.69	**1.00**	**1.00**	0.29	0.88	**1.00**	**1.00**	0.57	0.88	**1.00**	0.94	0.57	0.88	0.57	**0.94**	0.57	0.81	**1.00**	**1.00**	0.43	0.88
	F_1	0.93	0.86	**0.91**	0.25	0.90	0.93	**0.94**	0.25	0.93	0.67	**0.90**	0.50	0.76	0.93	**0.97**	0.44	0.93	0.93	**0.97**	0.73	0.90	0.93	**0.97**	0.67	0.90	0.67	**0.97**	0.62	0.79	0.93	**1.00**	0.50	0.85
20	Acc.	0.82	0.00	0.18	**0.91**	0.03	**0.64**	0.61	0.73	0.39	0.00	0.00	0.73	0.00	0.73	0.85	**1.00**	0.79	0.82	0.82	0.73	**0.88**	0.73	0.79	0.55	**0.88**	0.09	0.30	**0.91**	0.00	0.55	0.39	0.64	**0.97**
	F_1	0.89	0.00	0.31	**0.57**	0.06	**0.74**	0.71	0.47	0.19	0.00	0.00	0.59	0.00	0.84	**0.90**	0.61	0.87	**0.90**	0.86	0.67	0.62	0.84	0.87	0.55	**0.90**	0.17	0.45	**0.71**	0.00	**0.67**	0.57	0.45	0.27
23	Acc.	0.95	**1.00**	0.93	0.24	0.98	0.94	**0.95**	0.35	0.83	1.00	**0.98**	0.59	0.05	**1.00**	0.95	0.18	0.93	**1.00**	0.95	0.82	0.02	**1.00**	0.93	0.94	0.05	**1.00**	0.91	0.59	**1.00**	**1.00**	0.98	0.53	0.05
	F_1	0.90	**0.81**	0.74	0.38	0.70	**0.91**	0.83	0.44	0.40	0.67	**0.56**	0.69	0.05	0.92	0.92	0.30	**0.93**	**0.94**	0.91	0.85	0.05	**0.92**	0.91	0.84	0.09	**0.79**	0.75	0.71	0.62	**0.92**	0.80	0.56	0.07
30	Acc.	0.84	**0.80**	0.60	0.86	0.70	**0.93**	0.87	0.21	0.24	0.60	0.16	**0.93**	0.00	0.87	**0.97**	0.86	0.76	0.87	0.87	**0.93**	0.89	0.87	**0.89**	0.79	0.84	0.87	0.81	**0.93**	0.30	**0.87**	0.76	0.86	0.70
	F_1	0.84	**0.83**	0.68	0.71	0.48	**0.76**	0.67	0.32	0.36	0.64	0.22	**0.79**	0.00	**0.87**	0.77	0.80	0.78	0.81	**0.78**	0.74	0.75	0.87	**0.88**	0.73	0.75	**0.87**	0.73	0.84	0.27	**0.79**	0.68	0.77	0.68
33	Acc.	0.98	**1.00**	0.96	**1.00**	0.74	**1.00**	0.94	0.95	0.06	1.00	0.91	**1.00**	0.00	**1.00**	0.85	0.95	0.98	**1.00**	0.94	1.00	0.98	**1.00**	0.98	**1.00**	0.98	**1.00**	0.94	0.95	0.11	**1.00**	0.96	**1.00**	0.00
	F_1	0.98	0.97	0.96	**1.00**	0.84	**0.97**	0.96	0.92	0.11	0.97	0.94	**1.00**	0.00	0.97	0.91	0.95	**0.98**	0.97	0.95	**1.00**	0.98	0.97	**0.98**	0.97	0.98	0.95	0.96	**0.97**	0.20	0.97	0.97	**1.00**	0.00
34	Acc.	0.96	0.90	0.89	0.95	**0.96**	0.90	**0.96**	0.95	0.19	0.90	0.46	**1.00**	0.00	0.90	**1.00**	0.95	0.94	0.95	0.94	**1.00**	0.90	0.95	0.89	**1.00**	0.85	0.90	0.98	**1.00**	0.54	0.90	0.96	**1.00**	0.00
	F_1	0.96	0.94	0.91	**0.97**	0.96	0.94	0.94	**0.97**	0.32	0.94	0.89	**1.00**	0.00	0.94	0.93	**0.97**	0.95	**0.97**	0.94	**1.00**	0.93	0.97	0.92	**1.00**	0.91	0.94	0.96	**1.00**	0.70	0.94	0.95	**1.00**	0.00
46	Acc.	0.87	**1.00**	0.84	**1.00**	0.89	**1.00**	0.87	**1.00**	0.49	1.00	0.84	**1.00**	0.65	**1.00**	0.87	**1.00**	0.84	**1.00**	0.87	**1.00**	0.87	**1.00**	0.87	**1.00**	0.87	**1.00**	0.84	**1.00**	0.87	**1.00**	0.87	**1.00**	0.65
	F_1	0.91	0.90	0.89	**0.97**	0.69	**0.93**	0.88	0.79	0.38	0.85	0.90	**1.00**	0.14	**0.97**	0.91	0.94	0.90	0.93	0.90	**0.97**	0.93	0.93	0.91	**0.97**	0.91	0.93	0.87	**0.97**	0.48	0.93	0.89	**1.00**	0.56
48	Acc.	0.92	0.87	**0.92**	0.80	0.28	**0.80**	0.69	0.73	0.13	0.80	0.56	**0.67**	0.00	**0.87**	0.59	0.80	0.85	**0.87**	0.85	**0.87**	0.85	0.87	**0.90**	0.80	0.90	0.87	**0.90**	0.73	0.05	**0.93**	0.87	0.73	0.13
	F_1	0.88	**0.93**	0.78	0.89	0.43	**0.89**	0.76	0.76	0.22	0.89	0.58	**0.80**	0.00	**0.93**	0.74	0.86	0.87	**0.93**	0.87	**0.93**	0.72	**0.93**	**0.80**	0.89	0.74	**0.93**	0.86	0.85	0.10	**0.97**	0.88	0.85	0.23
all	WA	0.92	**0.86**	0.82	0.78	0.68	**0.87**	0.84	0.65	0.40	0.78	0.70	**0.83**	0.21	**0.93**	0.89	0.80	0.90	**0.93**	0.90	0.87	0.80	**0.93**	0.91	0.84	0.80	0.86	0.84	**0.87**	0.48	**0.89**	0.86	0.80	0.41

Table A2. Results on NTU-RGB+D dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/ Regression/training with occluded samples/reference case, Acc, F_1 and WA denote Accuracy, F_1 score and Weighted Accuracy, respectively. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". Numbers in italics denote the corresponding classes, as follows: 41: sneeze/cough, 42: staggering, 43: falling down, 44: headache, 45: chest pain, 46: back pain, 47: neck pain, 48: nausea/vomiting, 49: fan self, 103: yawn, 104: stretch oneself, 105: blow nose.

		None	**LA**				**RA**				**LA+RA**				**LL**				**RL**				**LL+RL**				**LA+LL**				**RA+RL**			
Class	**Metric**	**Bas.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**	**Gans**	**Reg.**	**Occ.**	**Ref.**
41	Acc.	0.89	0.74	0.70	**0.87**	0.73	0.32	0.51	**0.74**	0.14	0.16	0.57	**0.84**	0.11	0.74	0.73	**0.84**	0.24	**0.87**	0.65	0.81	0.62	**0.90**	0.87	0.74	0.24	0.77	0.70	**0.84**	0.16	0.16	**0.73**	0.71	0.11
	F_1	0.82	0.55	0.66	0.73	**0.74**	0.31	0.54	**0.65**	0.19	0.17	0.58	**0.79**	0.11	**0.72**	0.70	0.71	0.35	**0.86**	0.66	0.74	0.59	**0.86**	0.78	0.73	0.38	0.55	0.68	**0.80**	0.24	0.26	**0.73**	0.73	0.19
42	Acc.	0.35	0.28	0.22	**0.47**	0.13	**0.38**	0.17	0.34	0.26	**0.56**	0.26	0.41	0.00	**0.44**	0.22	0.41	0.09	0.34	0.26	**0.44**	0.09	**0.31**	0.30	0.44	0.00	0.28	0.13	**0.41**	0.00	**0.56**	0.22	0.47	0.09
	F_1	0.40	0.42	0.29	**0.42**	0.17	**0.46**	0.20	0.33	0.31	**0.48**	0.32	0.39	0.00	**0.53**	0.28	0.40	0.14	**0.46**	0.32	0.40	0.15	**0.41**	0.36	0.41	0.00	**0.39**	0.19	**0.39**	0.00	**0.55**	0.28	0.45	0.14
43	Acc.	0.66	**0.78**	0.62	0.59	0.72	**0.94**	0.66	0.53	0.83	0.50	**0.76**	0.69	0.62	**0.78**	0.69	0.47	0.52	**0.78**	0.76	0.69	0.76	**0.78**	0.76	0.69	0.72	0.69	**0.72**	0.69	0.62	0.69	**0.79**	0.69	0.55
	F_1	0.61	**0.56**	0.46	0.50	0.46	0.41	**0.47**	0.42	0.35	0.24	**0.49**	0.51	0.29	**0.62**	0.53	0.41	0.38	**0.68**	0.54	0.46	0.27	**0.71**	0.57	0.50	0.36	**0.55**	0.51	0.59	0.31	0.41	**0.56**	0.51	0.26
44	Acc.	0.93	**0.90**	0.87	0.87	0.77	0.87	**0.90**	0.87	0.07	0.71	**0.90**	0.87	0.07	0.84	**0.90**	0.87	0.93	0.84	0.90	0.90	**0.97**	0.94	0.90	0.84	**1.00**	0.94	0.90	0.87	**0.97**	0.74	**0.90**	0.84	0.37
	F_1	0.95	**0.95**	0.90	0.89	0.87	**0.93**	**0.93**	0.89	0.12	0.83	**0.95**	0.89	0.12	0.90	**0.93**	0.87	0.93	0.90	0.95	0.90	**0.95**	**0.95**	0.93	0.88	0.85	**0.97**	0.93	0.87	0.89	0.85	**0.93**	0.88	0.54
45	Acc.	0.41	**0.52**	0.18	0.28	0.46	0.03	0.23	0.19	**0.91**	0.32	0.23	0.28	**0.41**	**0.45**	0.23	0.25	0.14	**0.61**	0.23	0.34	0.59	**0.52**	0.50	0.22	0.50	**0.45**	0.18	0.41	0.14	0.23	0.41	0.16	**0.77**
	F_1	0.51	**0.62**	0.22	0.27	0.40	0.06	0.24	0.23	**0.26**	0.05	0.23	0.25	**0.23**	**0.57**	0.26	0.25	0.17	**0.68**	0.26	0.29	0.32	**0.63**	0.47	0.26	0.23	**0.57**	0.21	0.38	0.19	0.31	0.40	0.19	**0.23**
46	Acc.	0.38	0.44	0.32	**0.47**	0.24	0.16	0.27	**0.47**	0.12	0.00	0.35	**0.56**	0.27	**0.56**	0.50	0.53	0.32	**0.56**	0.29	0.44	0.15	**0.50**	0.29	0.50	0.09	**0.53**	0.35	0.59	0.12	0.44	0.35	**0.66**	0.03
	F_1	0.43	**0.49**	0.39	0.35	0.31	0.23	0.30	**0.33**	0.17	0.00	0.39	**0.41**	0.31	**0.59**	0.47	0.36	0.37	**0.57**	0.33	0.38	0.21	**0.50**	0.35	0.36	0.14	**0.55**	0.37	0.41	0.20	0.42	0.40	**0.42**	0.05
47	Acc.	0.53	**0.65**	0.40	0.42	0.63	**0.71**	0.27	0.42	0.27	**0.77**	0.27	0.58	0.07	**0.58**	0.33	0.45	0.33	**0.65**	0.27	0.52	0.37	**0.65**	0.33	0.65	0.13	0.58	0.40	**0.71**	0.33	**0.81**	0.33	0.65	0.43
	F_1	0.64	**0.74**	0.53	0.55	0.73	**0.79**	0.39	0.55	0.28	**0.69**	0.38	0.65	0.11	**0.69**	0.45	0.56	0.43	**0.73**	0.39	0.58	0.48	**0.77**	0.45	0.63	0.21	**0.72**	0.51	0.70	0.43	**0.72**	0.45	0.63	0.36
48	Acc.	0.79	0.41	0.38	0.10	**0.42**	0.28	**0.38**	0.07	0.17	**0.62**	0.38	0.03	0.50	0.50	0.54	0.07	**0.67**	**0.66**	0.29	0.10	0.21	**0.50**	0.50	0.10	0.21	0.47	0.38	0.07	**0.63**	0.22	**0.38**	0.03	0.21
	F_1	0.55	0.43	0.31	0.15	**0.47**	0.30	**0.32**	0.10	0.18	0.11	**0.31**	0.06	0.14	0.52	**0.39**	0.11	0.31	**0.58**	0.24	0.16	0.22	**0.46**	0.44	0.17	0.14	**0.54**	0.32	0.12	0.32	0.29	**0.32**	0.06	0.15
49	Acc.	0.68	**0.53**	0.48	0.44	0.48	0.25	**0.46**	0.38	0.05	0.19	0.36	**0.50**	0.00	**0.78**	0.46	0.28	0.77	**0.69**	0.48	0.56	0.16	**0.69**	0.48	0.44	0.41	0.53	0.46	0.38	**0.59**	0.25	**0.52**	0.41	0.23
	F_1	0.59	**0.51**	0.42	0.44	0.47	0.34	**0.41**	0.39	0.07	0.23	0.36	**0.42**	0.00	**0.60**	0.44	0.35	0.46	**0.61**	0.42	0.47	0.24	**0.59**	0.45	0.38	0.31	**0.49**	0.43	0.40	0.40	0.25	**0.46**	0.37	0.26
103	Acc.	0.79	**0.81**	0.76	0.72	0.79	0.56	**0.82**	0.78	0.40	0.63	**0.76**	0.53	0.16	0.72	**0.76**	0.75	0.74	0.75	**0.84**	0.50	0.74	**0.84**	0.82	0.69	0.61	**0.81**	0.79	0.72	0.74	0.59	**0.79**	0.72	0.32
	F_1	0.85	**0.87**	0.79	0.81	0.79	0.68	**0.84**	0.83	0.54	0.68	**0.79**	0.68	0.26	0.79	0.82	0.81	**0.84**	0.83	0.84	0.67	**0.85**	**0.89**	0.83	0.77	0.72	**0.87**	0.85	0.79	0.82	0.68	**0.83**	0.81	0.48
104	Acc.	0.97	0.84	**0.94**	0.75	0.74	0.78	**0.94**	0.75	0.21	0.00	**0.88**	0.69	0.00	**0.97**	0.88	0.84	0.21	**0.94**	**0.94**	0.63	0.15	**0.94**	0.88	0.78	0.03	0.91	**0.94**	0.78	0.12	0.56	**0.91**	0.81	0.32
	F_1	0.96	0.90	**0.93**	0.81	0.82	0.85	**0.90**	0.76	0.34	0.00	**0.88**	0.79	0.00	**0.97**	0.90	0.84	0.33	**0.95**	0.93	0.75	0.24	**0.95**	0.90	0.86	0.06	0.88	**0.91**	0.83	0.21	0.71	**0.89**	0.85	0.48
105	Acc.	0.60	**0.78**	0.66	0.34	0.77	**0.69**	0.51	0.38	0.11	0.59	**0.63**	0.22	0.00	**0.81**	0.57	0.47	0.14	**0.84**	0.60	0.16	0.11	**0.84**	0.60	0.25	0.00	**0.75**	0.69	0.28	0.20	**0.66**	0.54	0.28	0.03
	F_1	0.66	**0.67**	0.64	0.37	0.65	0.52	**0.54**	0.38	0.17	0.34	**0.65**	0.27	0.00	**0.70**	0.61	0.46	0.24	**0.68**	0.62	0.22	0.19	**0.67**	0.65	0.31	0.00	**0.65**	0.69	0.32	0.26	0.46	**0.60**	0.35	0.05
all	WA	0.68	**0.64**	0.57	0.53	0.59	0.50	**0.53**	0.49	0.27	0.35	**0.55**	0.52	0.16	**0.68**	0.59	0.52	0.44	**0.71**	0.57	0.51	0.41	**0.70**	0.62	0.53	0.33	**0.64**	0.58	0.56	0.40	0.49	**0.59**	0.53	0.25

Table 3. Results on UTKinect-Action3D dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/Regression/training with occluded samples/reference case, Acc, F_1 and WA denote Accuracy, F_1 score and Weighted Accuracy, respectively. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". Numbers in italics denote the corresponding classes, as follows: 0: walk, 1: sitDown, 2: standUp, 3: pickUp, 4: carry, 5: throw, 6: push, 7: pull, 8: wavehands, 9: clapHands.

		None	LA				RA				LA+RA				LL				RL				LL+RL				LA+LL				RA+RL			
Class	**Metric**	**Baseline**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**
0	Acc.	0.50	**0.50**	0.40	0.30	0.30	**1.00**	0.40	0.30	0.00	**0.50**	0.40	0.30	0.00	**1.00**	0.40	0.30	0.20	**1.00**	0.40	0.40	0.50	**1.00**	0.40	0.30	0.20	0.50	0.40	0.50	**0.70**	**1.00**	0.40	0.40	0.00
	F_1	0.48	**0.50**	0.43	0.35	0.31	**0.80**	0.43	0.35	0.00	**0.67**	0.39	0.26	0.00	**1.00**	0.43	0.31	0.09	**0.80**	0.43	0.41	0.41	**1.00**	0.43	0.28	0.07	**0.67**	0.43	0.44	0.33	**1.00**	0.39	0.43	0.00
1	Acc.	0.70	0.00	0.30	0.20	**0.60**	**0.50**	0.10	0.20	0.30	0.00	0.30	0.20	**0.90**	**1.00**	0.70	0.60	0.00	**0.50**	**0.50**	0.30	0.40	0.50	**0.60**	0.10	0.00	0.00	**0.40**	0.50	0.20	0.00	**0.30**	0.30	0.20
	F_1	0.53	0.00	0.23	0.16	**0.15**	**0.50**	0.08	0.10	0.17	0.00	0.13	0.11	**0.18**	**0.80**	0.57	0.51	0.00	0.50	**0.39**	0.26	0.35	0.50	**0.43**	0.10	0.00	0.00	**0.32**	0.39	0.06	0.00	**0.21**	0.23	0.13
2	Acc.	0.80	**1.00**	0.80	0.70	0.00	**1.00**	0.80	0.50	0.00	**1.00**	0.90	0.80	0.00	**1.00**	0.80	0.80	0.10	**1.00**	0.70	0.80	0.60	**1.00**	0.70	0.60	0.00	**1.00**	0.90	0.90	0.20	**1.00**	0.90	0.40	0.00
	F_1	0.87	**1.00**	0.76	0.66	0.00	0.67	**0.80**	0.41	0.00	0.80	**0.83**	0.76	0.00	**0.80**	0.80	0.67	0.13	**0.80**	0.75	0.66	0.63	**0.80**	0.75	0.46	0.00	0.80	0.79	**0.85**	0.27	**1.00**	0.89	0.26	0.00
3	Acc.	1.00	**0.50**	0.30	0.30	0.00	0.50	**0.60**	0.10	0.20	0.00	0.00	**0.30**	0.00	**1.00**	0.70	0.40	0.40	0.00	0.70	0.60	**1.00**	**1.00**	0.80	0.30	0.70	0.00	0.20	**0.70**	0.00	0.00	0.20	**0.30**	**0.30**
	F_1	0.92	**0.50**	0.22	0.26	0.00	0.40	**0.43**	0.05	0.05	0.00	0.00	**0.33**	0.00	**1.00**	0.58	0.41	0.40	0.00	0.63	0.57	**0.69**	0.67	**0.72**	0.18	0.61	0.00	0.11	**0.63**	0.00	0.00	**0.16**	0.29	0.07
4	Acc.	0.90	0.00	0.00	**0.20**	0.00	0.00	0.40	0.10	**0.60**	0.00	0.00	**0.50**	0.00	0.50	0.60	0.20	**0.90**	**1.00**	0.60	0.60	0.50	**1.00**	**0.60**	0.10	**0.80**	**1.00**	0.10	0.50	0.00	0.50	0.30	0.10	**0.60**
	F_1	0.93	0.00	0.00	**0.11**	0.00	0.00	**0.40**	0.13	0.14	0.00	0.00	**0.41**	0.00	0.50	0.57	0.13	**0.73**	**0.57**	0.60	0.57	0.39	**1.00**	**0.57**	0.08	**0.59**	**0.50**	0.08	0.46	0.00	**0.40**	0.20	0.13	0.14
5	Acc.	1.00	**1.00**	0.80	0.50	0.00	0.00	**0.80**	0.50	0.00	0.00	**0.80**	0.40	0.00	0.00	0.80	0.50	**0.90**	0.50	**0.80**	0.40	0.70	0.50	**0.80**	0.50	**0.80**	0.00	**0.80**	0.50	0.10	**1.00**	0.80	0.50	0.00
	F_1	0.92	**1.00**	0.69	0.49	0.00	0.00	**0.69**	0.43	0.00	0.00	**0.72**	0.47	0.00	0.00	**0.69**	0.50	0.58	0.67	**0.69**	0.43	0.65	0.67	**0.69**	0.40	0.59	0.00	**0.73**	0.60	0.13	**1.00**	0.76	0.43	0.00
6	Acc.	1.00	**1.00**	0.80	0.50	0.00	1.00	**0.80**	0.70	0.00	**1.00**	0.90	0.80	0.00	**1.00**	0.90	0.80	**1.00**	**1.00**	0.80	0.80	0.90	**1.00**	0.80	0.70	0.80	**1.00**	0.80	1.00	0.20	**1.00**	1.00	0.70	0.00
	F_1	1.00	**1.00**	0.80	0.49	0.00	1.00	**0.80**	0.66	0.00	**0.80**	0.93	0.61	0.00	0.80	**0.93**	0.72	0.79	**1.00**	0.76	0.68	0.81	**1.00**	0.76	0.63	0.51	0.80	0.83	**0.86**	0.13	**1.00**	0.96	0.63	0.00
7	Acc.	0.40	0.00	0.50	**0.60**	**0.70**	0.00	**0.40**	0.30	0.00	**1.00**	0.20	0.60	0.00	0.00	**0.40**	0.30	0.00	0.00	**0.20**	0.70	0.10	0.00	0.30	**0.40**	0.00	0.00	**0.40**	0.30	**0.40**	**0.50**	0.40	**0.50**	0.00
	F_1	0.43	0.00	**0.25**	0.22	0.23	0.00	**0.28**	0.21	0.00	**0.36**	0.17	0.35	0.00	0.00	**0.40**	0.19	0.00	0.00	**0.12**	0.54	0.10	0.00	0.27	**0.34**	0.00	0.00	**0.24**	0.21	0.11	**0.67**	0.34	0.28	0.00
8	Acc.	0.60	**1.00**	0.40	0.50	0.00	1.00	**0.40**	0.30	0.00	**1.00**	0.20	0.50	0.00	**1.00**	0.40	0.60	0.50	**1.00**	0.40	0.60	0.40	**1.00**	0.40	0.50	0.30	**1.00**	0.30	0.50	0.20	**1.00**	0.40	0.40	0.00
	F_1	0.52	**0.80**	0.36	0.44	0.00	1.00	**0.36**	0.29	0.00	**1.00**	**0.16**	0.47	0.00	**1.00**	0.27	0.55	0.40	**1.00**	0.36	0.55	0.43	**1.00**	0.29	0.53	0.27	**0.80**	0.26	0.53	0.27	**1.00**	0.36	0.44	0.00
9	Acc.	1.00	**1.00**	0.70	0.40	0.00	1.00	**1.00**	0.30	0.40	**1.00**	0.30	1.00	0.00	**1.00**	**1.00**	0.70	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	1.00	0.90	**1.00**	**1.00**	0.80	**1.00**	0.20	**1.00**	1.00	0.80	0.00
	F_1	0.87	**0.57**	0.52	0.23	0.00	1.00	**0.83**	0.10	0.47	**1.00**	0.13	0.88	0.00	**1.00**	0.87	0.59	0.69	**1.00**	0.87	**0.96**	0.96	**1.00**	0.87	0.81	0.74	**1.00**	0.58	**0.92**	0.07	**1.00**	0.87	0.76	0.00
all	WA	0.79	**0.60**	0.50	0.42	0.16	0.60	**0.57**	0.33	0.15	**0.55**	0.40	0.54	0.09	**0.75**	0.67	0.52	0.50	**0.70**	0.61	0.62	0.61	**0.80**	0.64	0.44	0.46	0.55	0.51	**0.64**	0.22	**0.70**	0.57	0.44	0.11

Table 4. Results on SYSU-3D-HOI dataset. "Bas."/"Gan"/"Reg."/"Occ."/"Ref." denote baseline/Generative adversarial networks/ Regression/training with occluded samples/reference case, Acc, F_1 and WA denote Accuracy, F_1 score and Weighted Accuracy, respectively. "None" denotes the case without occlusion. LA, RA, LL, RL denote the occlusion of left arm, right arm, left leg, right leg, respectively. Numbers in bold indicate best performance between "Rec."/"Occ."/"Ref.". Numbers in italics denote the corresponding classes, as follows: 1: drinking, 2: pouring, 3: calling phone, 4: playing phone, 5: wearing backpacks, 6: packing backpacks, 7: sitting chair, 8: moving chair, 9: taking out wallet, 10: taking from wallet, 11: mopping, 12: sweeping.

		None	**LA**				**RA**				**LA+RA**				**LL**				**RL**				**LL+RL**				**LA+LL**				**RA+RL**			
Class	**Metric**	**Baseline**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**	**Gan**	**Reg.**	**Occ.**	**Ref.**
1	Acc.	0.10	0.00	0.00	**0.13**	0.00	0.00	0.05	**0.13**	0.00	**0.25**	0.00	**0.25**	0.00	0.00	0.00	**0.10**	0.00	0.00	0.10	**0.28**	0.00	0.00	0.05	**0.13**	0.00	**0.00**	0.00	**0.28**	0.00	0.00	0.00	**0.05**	0.00
	F_1	0.10	0.00	0.00	**0.14**	0.00	0.00	0.06	**0.07**	0.00	0.22	0.00	**0.25**	0.00	0.00	0.00	**0.11**	0.00	0.00	0.10	**0.28**	0.00	0.00	0.08	**0.11**	0.00	0.00	0.00	**0.28**	0.00	0.00	0.00	**0.06**	0.00
2	Acc.	0.65	0.25	**0.45**	0.43	0.00	0.25	**0.65**	0.13	0.05	0.25	0.00	**0.50**	0.00	**0.50**	0.45	0.40	0.45	0.25	**0.55**	0.53	0.20	0.50	**0.55**	0.28	0.10	**0.50**	**0.50**	0.48	0.30	0.25	0.00	**0.28**	0.00
	F_1	0.59	0.22	0.36	**0.46**	0.00	0.33	**0.63**	0.12	0.08	0.40	0.00	**0.54**	0.00	0.40	0.44	**0.45**	0.34	0.33	0.56	**0.57**	0.23	**0.44**	**0.53**	0.34	0.04	0.33	**0.48**	0.49	0.16	0.40	0.00	**0.19**	0.00
3	Acc.	0.80	0.50	**0.75**	0.60	0.55	0.75	**0.85**	0.48	0.05	0.50	0.60	**0.68**	0.45	0.50	**0.70**	0.58	0.00	0.50	**0.85**	0.68	0.00	0.25	**0.80**	0.48	0.00	0.50	0.60	**0.65**	0.25	0.25	**0.55**	0.55	0.10
	F_1	0.77	0.67	**0.69**	0.54	0.21	0.67	**0.77**	0.44	0.05	**0.67**	0.57	0.59	0.12	0.57	**0.65**	0.56	0.00	0.67	**0.71**	0.59	0.00	0.33	**0.73**	0.46	0.00	**0.57**	0.56	0.59	0.18	0.33	**0.61**	0.50	0.06
4	Acc.	0.75	**0.75**	0.70	0.58	0.05	0.75	**0.80**	0.53	0.30	**0.75**	**0.75**	0.65	0.00	**0.75**	**0.75**	0.68	0.40	**0.75**	**0.75**	0.65	0.25	**0.75**	0.70	0.58	0.20	**0.75**	**0.75**	0.70	0.25	0.75	**0.80**	0.68	0.25
	F_1	0.80	**0.75**	0.73	0.61	0.08	**0.86**	0.84	0.57	0.43	**0.86**	0.76	0.71	0.00	**0.86**	0.79	0.70	0.52	**0.86**	0.78	0.72	0.34	**0.86**	0.75	0.62	0.29	**0.75**	**0.75**	0.76	0.27	**0.86**	0.76	0.66	0.31
5	Acc.	0.55	**0.50**	0.45	0.28	0.10	**1.00**	0.40	0.53	0.40	**0.75**	0.50	0.40	0.00	**0.75**	0.45	0.38	0.40	**1.00**	0.40	0.40	0.00	**0.75**	0.50	0.25	0.20	**0.75**	0.30	0.33	0.50	**1.00**	0.50	0.30	0.10
	F_1	0.48	**0.50**	0.34	0.21	0.10	**0.80**	0.35	0.40	0.15	**0.67**	0.30	0.38	0.00	**0.67**	0.38	0.32	0.10	**0.80**	0.39	0.41	0.00	**0.67**	0.39	0.23	0.19	**0.86**	0.28	0.32	0.20	**0.67**	0.44	0.31	0.07
6	Acc.	0.40	**0.75**	0.25	0.28	0.05	**0.50**	0.25	0.05	0.00	0.00	0.00	**0.30**	0.00	**0.50**	0.30	0.33	0.00	**0.75**	0.35	0.28	0.00	**0.75**	0.25	0.25	0.00	**0.50**	0.30	0.33	0.00	**0.50**	0.25	0.43	0.00
	F_1	0.38	**0.60**	0.31	0.27	0.08	**0.50**	0.28	0.03	0.00	0.00	0.00	**0.28**	0.00	**0.40**	0.27	0.31	0.00	**0.60**	0.32	0.26	0.00	**0.60**	0.18	0.22	0.00	**0.40**	0.27	0.30	0.00	**0.40**	0.20	0.38	0.00
7	Acc.	0.45	0.25	0.40	0.43	**0.50**	0.25	**0.35**	0.33	0.30	0.00	0.15	**0.65**	0.00	0.25	0.40	**0.43**	0.00	0.25	**0.45**	0.68	0.00	0.25	**0.45**	0.45	0.00	0.25	0.35	**0.55**	0.20	0.25	0.35	**0.53**	0.00
	F_1	0.54	**0.40**	0.35	0.39	0.18	**0.40**	0.36	0.29	0.19	0.00	0.20	**0.61**	0.00	0.40	**0.47**	0.42	0.00	0.29	**0.46**	0.60	0.00	0.29	**0.42**	0.42	0.00	0.40	0.41	**0.52**	0.06	0.33	0.40	**0.43**	0.00
8	Acc.	0.90	**1.00**	0.85	0.80	0.10	**1.00**	0.85	0.75	0.65	**1.00**	0.75	0.75	0.00	**1.00**	0.90	0.78	0.80	**1.00**	0.95	0.80	0.60	**1.00**	0.95	0.73	0.60	**1.00**	0.90	0.78	0.50	**1.00**	0.85	0.80	0.35
	F_1	0.88	**1.00**	0.81	0.79	0.13	**1.00**	0.83	0.74	0.71	**1.00**	0.77	0.77	0.00	**1.00**	0.84	0.79	0.75	**0.89**	0.87	0.79	0.58	**1.00**	0.88	0.74	0.63	**1.00**	0.88	0.81	0.56	**1.00**	0.80	0.84	0.46
9	Acc.	0.65	**1.00**	0.30	0.35	0.50	**0.75**	0.25	0.30	0.40	**1.00**	0.20	0.33	0.80	**1.00**	0.40	0.40	0.00	**1.00**	0.35	0.38	0.00	**1.00**	0.35	0.28	0.00	**1.00**	0.40	0.35	0.10	**1.00**	0.25	0.30	0.65
	F_1	0.67	**0.62**	0.39	0.31	0.19	**0.60**	0.31	0.28	0.31	**0.73**	0.28	0.34	0.19	**0.73**	0.47	0.30	0.00	**0.80**	0.45	0.39	0.00	**0.67**	0.38	0.24	0.00	**0.80**	0.44	0.31	0.12	**0.67**	0.26	0.25	0.12
10	Acc.	0.30	0.00	0.15	**0.33**	0.00	0.25	**0.30**	0.13	0.05	0.00	0.00	0.28	0.00	0.00	0.30	**0.40**	0.00	0.25	**0.30**	0.28	0.00	0.25	0.25	**0.33**	0.00	0.25	**0.30**	0.30	0.05	**0.50**	0.25	0.15	0.05
	F_1	0.24	0.00	0.12	**0.27**	0.00	**0.22**	0.21	0.07	0.06	0.00	0.00	0.22	0.00	0.00	0.20	**0.29**	0.00	0.20	**0.22**	0.21	0.00	0.25	0.18	**0.26**	0.00	0.29	**0.31**	0.24	0.06	**0.36**	0.20	0.12	0.08
11	Acc.	0.45	**0.50**	0.40	0.30	0.00	**0.50**	0.45	0.25	0.10	**0.50**	0.30	0.35	0.00	0.50	**0.60**	0.20	0.00	**0.50**	**0.50**	0.33	0.15	0.50	**0.55**	0.33	0.00	**0.50**	0.25	0.28	0.00	0.25	**0.50**	0.23	0.00
	F_1	0.34	**0.31**	0.31	0.24	0.00	0.31	**0.32**	0.14	0.10	**0.33**	0.23	0.26	0.00	0.36	**0.43**	0.15	0.00	0.40	**0.44**	0.25	0.10	0.36	**0.39**	0.31	0.00	**0.31**	0.17	0.21	0.00	0.18	**0.29**	0.19	0.00
12	Acc.	0.45	0.25	0.35	**0.40**	0.00	0.00	0.25	0.40	**0.35**	0.25	**0.55**	0.23	0.00	0.25	0.30	0.35	**0.40**	0.00	0.20	0.35	**0.90**	0.00	0.10	0.38	**0.80**	0.00	**0.50**	0.23	0.20	0.00	**0.35**	0.20	0.10
	F_1	0.45	0.40	0.34	**0.41**	0.00	0.00	0.22	**0.45**	0.18	0.20	**0.24**	0.24	0.00	0.29	**0.25**	0.39	0.09	0.00	0.15	0.38	**0.18**	0.00	0.10	0.32	0.16	0.00	**0.38**	0.25	0.14	0.00	**0.23**	0.19	0.02
all	WA	0.54	**0.48**	0.42	0.41	0.15	**0.50**	0.45	0.33	0.22	0.44	0.32	**0.45**	0.10	**0.50**	0.46	0.42	0.20	**0.52**	0.48	0.47	0.18	**0.50**	0.46	0.37	0.16	**0.50**	0.43	0.44	0.20	**0.48**	0.39	0.37	0.13

References

1. Wang, P.; Li, W.; Ogunbona, P.; Wan, J.; Escalera, S. RGB-D-based human motion recognition with deep learning: A survey. *Comput. Vis. Image Underst.* **2018**, *171*, 118–139. [CrossRef]
2. Papadakis, A.; Mathe, E.; Spyrou, E.; Mylonas, P. A geometric approach for cross-view human action recognition using deep learning. In Proceedings of the 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), Dubrovnik, Croatia, 23–25 September 2019; pp. 258–263.
3. Zhang, Z. Microsoft kinect sensor and its effect. *IEEE Multimed.* **2012**, *19*, 4–10. [CrossRef]
4. Peng, K.; Roitberg, A.; Yang, K.; Zhang, J.; Stiefelhagen, R. Delving deep into one-shot skeleton-based action recognition with diverse occlusions. *IEEE Trans. Multimed.* **2023**, *25*, 1489–1504. [CrossRef]
5. Kim, J.; Shin, W.; Park, H.; Baek, J. Addressing the occlusion problem in multi-camera people tracking with human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 5463–5469.
6. Ahad, M.; Tan, J.; Kim, H.; Ishikawa, S. Solutions to motion self-occlusion problem in human activity analysis. In Proceedings of the 11th International Conference on Computer and Information Technology, Khulna, Bangladesh, 24–27 December 2008.
7. Giannakos, I.; Mathe, E.; Spyrou, E.; Mylonas, P. A study on the Effect of Occlusion in Human Activity Recognition. In Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference, Corfu Greece, 29 June–2 July 2021; pp. 473–482.
8. Vernikos, I.; Spyrou, E.; Kostis, I.A.; Mathe, E.; Mylonas, P. A deep regression approach for human activity recognition under partial occlusion. *Int. J. Neural Syst.* **2023**, *33*, 2350047. [CrossRef] [PubMed]
9. Du, Y.; Fu, Y.; Wang, L. Skeleton based action recognition with convolutional neural network. In Proceedings of the 2015 3rd IAPR Asian conference on pattern recognition (ACPR), Kuala Lumpur, Malaysia, 3–6 November 2015; pp. 579–583.
10. Wang, P.; Li, Z.; Hou, Y.; Li, W. Action recognition based on joint trajectory maps using convolutional neural networks. In Proceedings of the 24th ACM international conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 102–106.
11. Hou, Y.; Li, Z.; Wang, P.; Li, W. Skeleton optical spectra-based action recognition using convolutional neural networks. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *28*, 807–811. [CrossRef]
12. Li, C.; Hou, Y.; Wang, P.; Li, W. Joint distance maps based action recognition with convolutional neural networks. *IEEE Signal Process. Lett.* **2017**, *24*, 624–628. [CrossRef]
13. Liu, M.; Liu, H.; Chen, C. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognit.* **2017**, *68*, 346–362. [CrossRef]
14. Ke, Q.; An, S.; Bennamoun, M.; Sohel, F.; Boussaid, F. Skeletonnet: Mining deep part features for 3-d action recognition. *IEEE Signal Process. Lett.* **2017**, *24*, 731–735. [CrossRef]
15. Iosifidis, A.; Tefas, A.; Pitas, I. Multi-view human action recognition under occlusion based on fuzzy distances and neural networks. In Proceedings of the 20th European Signal Processing Conference (EUSIPCO), Bucharest, Romania, 27–31 August 2012; pp. 1129–1133.
16. Angelini, F.; Fu, Z.; Long, Y.; Shao, L.; Naqvi, S.M. 2D pose-based real-time human action recognition with occlusion-handling. *IEEE Trans. Multimed.* **2019**, *22*, 1433–1446. [CrossRef]
17. Li, W.; Zhang, Z.; Liu, Z. Action recognition based on a bag of 3d points. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, San Francisco, CA, USA, 13–18 June 2010; pp. 9–14.
18. Kim, M.H.; Kim, M.J.; Yoo, S.B. Occluded Part-aware Graph Convolutional Networks for Skeleton-based Action Recognition. In Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; pp. 7310–7317.
19. Chen, Z.; Wang, H.; Gui, J. Occluded Skeleton-Based Human Action Recognition with Dual Inhibition Training. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, ON, Canad, 29 October–3 November 2023; pp. 2625–2634.
20. Gu, R.; Wang, G.; Hwang, J.N. Exploring severe occlusion: Multi-person 3d pose estimation with gated convolution. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 8243–8250.
21. Yang, D.; Wang, Y.; Dantcheva, A.; Garattoni, L.; Francesca, G.; Brémond, F. Self-Supervised Video Pose Representation Learning for Occlusion-Robust Action Recognition. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–5.
22. Ma, L.; Huang, K.; Wei, D.; Ming, Z.; Shen, H. Fda-gan: Flow-based dual attention gan for human pose transfer. *IEEE Trans. Multimed.* **2021**, *25*, 930–941. [CrossRef]
23. Hernandez, A.; Gall, J.; Moreno-Noguer, F. Human motion prediction via spatio-temporal inpainting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7134–7143.
24. Lee, W.; Park, S.; Kim, T. Denoising Graph Autoencoder for Missing Human Joints Reconstruction. *IEEE Access* **2024**, *12*, 57381–57389. [CrossRef]

25. Irsch, M.P. Semi-Supervised Human Activity Recognition with the Wearable Respeck Sensor Using GANs. Master's Thesis, School of Informatics, University of Edinburgh, Edinburgh, UK, 2019.
26. Li, X.A.; Luo, J.; Younes, R. ActivityGAN: Generative adversarial networks for data augmentation in sensor-based human activity recognition. In Proceedings of the Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, Virtual, 12–17 September 2020; pp. 249–254.
27. Abedin, A.; Rezatofighi, H.; Ranasinghe, D.C. Guided-GAN: Adversarial Representation Learning for Activity Recognition with Wearables. *arXiv* **2021**, arXiv:2110.05732.
28. Rahman, M.M.; Gurbuz, S.Z.; Amin, M.G. Physics-aware generative adversarial networks for radar-based human activity recognition. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *59*, 2994–3008. [CrossRef]
29. Soleimani, E.; Nazerfard, E. Cross-subject transfer learning in human activity recognition systems using generative adversarial networks. *Neurocomputing* **2021**, *426*, 26–34. [CrossRef]
30. Jimale, A.O.; Noor, M.H.M. Fully connected generative adversarial network for human activity recognition. *IEEE Access* **2022**, *10*, 100257–100266. [CrossRef]
31. Qu, L.; Wang, Y.; Yang, T.; Sun, Y. Human activity recognition based on WRGAN-GP-synthesized micro-Doppler spectrograms. *IEEE Sens. J.* **2022**, *22*, 8960–8973. [CrossRef]
32. Wang, J.; Chen, Y.; Gu, Y.; Xiao, Y.; Pan, H. Sensorygans: An effective generative adversarial framework for sensor-based human activity recognition. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.
33. Gammulle, H.; Denman, S.; Sridharan, S.; Fookes, C. Multi-level sequence GAN for group activity recognition. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018*; Revised Selected Papers, Part I 14; Springer International Publishing: Basel, Switzerland, 2019; pp. 331–346.
34. Erol, B.; Gurbuz, S.Z.; Amin, M.G. GAN-based synthetic radar micro-Doppler augmentations for improved human activity recognition. In Proceedings of the 2019 IEEE Radar Conference (RadarConf), Boston, MA, USA, 22–26 April 2019; pp. 1–5.
35. Zadeh, M.Z.; Ramesh Babu, A.; Jaiswal, A.; Kyrarini, M.; Makedon, F. Self-supervised human activity recognition by augmenting generative adversarial networks. In Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference, Corfu, Greece, 29 June–2 July 2021; pp. 171–176.
36. Wang, D.; Yang, J.; Cui, W.; Xie, L.; Sun, S. Multimodal CSI-based human activity recognition using GANs. *IEEE Internet Things J.* **2021**, *8*, 17345–17355. [CrossRef]
37. Hasan, K.; Uddin, M.Z.; Ray, A.; Hasan, M.; Alnajjar, F.; Ahad, M.A.R. Improving Gait Recognition through Occlusion Detection and Silhouette Sequence Reconstruction. *IEEE Access* **2024**, *12*, 158597–158610. [CrossRef]
38. Xu, L.; Song, Z.; Wang, D.; Su, J.; Fang, Z.; Ding, C.; Gan, W.; Yan, Y.; Jin, X.; Yang, X.; et al. Actformer: A gan-based transformer towards general action-conditioned 3d human motion generation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–3 October 2023; pp. 2228–2238.
39. Fukushi, K.; Nozaki, Y.; Nishihara, K.; Nakahara, K. Few-shot generative model for skeleton-based human action synthesis using cross-domain adversarial learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 3946–3955.
40. Song, S.; Liu, J.; Lin, L.; Guo, Z. Learning to recognize human actions from noisy skeleton data via noise adaptation. *IEEE Trans. Multimed.* **2021**, *24*, 1152–1163. [CrossRef]
41. Zalameda, J.G.; Kruse, B.; Glandon, A.M.; Witherow, M.A.; Shetty, S.; Iftekharuddin, K.M. Generalized Adversarial and Hierarchical Co-occurrence Network based Synthetic Skeleton Generation and Human Identity Recognition. In Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18–23 July 2022; pp. 1–8.
42. Degardin, B.; Neves, J.; Lopes, V.; Brito, J.; Yaghoubi, E.; Proença, H. Generative adversarial graph convolutional networks for human action synthesis. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 1150–1159.
43. Avola, D.; Cascio, M.; Cinque, L.; Foresti, G.L.; Massaroni, C.; Rodolà, E. 2-D skeleton-based action recognition via two-branch stacked LSTM-RNNs. *IEEE Trans. Multimed.* **2019**, *22*, 2481–2496. [CrossRef]
44. Shen, J.; Dudley, J.; Kristensson, P.O. The imaginative generative adversarial network: Automatic data augmentation for dynamic skeleton-based hand gesture and human action recognition. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–8.
45. Pan, Q.; Zhao, Z.; Xie, X.; Li, J.; Cao, Y.; Shi, G. View-normalized skeleton generation for action recognition. In Proceedings of the 29th ACM International Conference on Multimedia, Chengdu, China, 20–24 October 2021; pp. 1875–1883.
46. Liu, L.; Wang, X.; Li, F.; Chen, J. Active Generation Network of Human Skeleton for Action Recognition. *arXiv* **2024**, arXiv:2401.17086.
47. Li, Z.; Li, D. Action recognition of construction workers under occlusion. *J. Build. Eng.* **2022**, *45*, 103352. [CrossRef]

48. Papadakis, A.; Mathe, E.; Vernikos, I.; Maniatis, A.; Spyrou, E.; Mylonas, P. Recognizing human actions using 3d skeletal information and CNNs. In *Engineering Applications of Neural Networks: 20th International Conference, EANN 2019, Xersonisos, Crete, Greece, 24–26 May 2019*; Proceedings 20; Springer: Cham, Switzerland, 2019.
49. Vernikos, I.; Mathe, E.; Papadakis, A.; Spyrou, E.; Mylonas, P. An image representation of skeletal data for action recognition using convolutional neural networks. In Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments, Rhodes, Greece, 5–7 June 2019; pp. 325–326.
50. Mathe, E.; Mitsou, A.; Spyrou, E.; Mylonas, P. Arm Gesture Recognition using a Convolutional Neural Network. In Proceedings of the 2018 13th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), Zaragoza, Spain, 6–7 September 2018.
51. Mathe, E.; Maniatis, A.; Spyrou, E.; Mylonas, P. A deep learning approach for human action recognition using skeletal information. In *GeNeDis 2018: Computational Biology and Bioinformatics*; Springer International Publishing: Basel, Switzerland, 2020; pp. 105–114.
52. Spyrou, E.; Mathe, E.; Pikramenos, G.; Kechagias, K.; Mylonas, P. Data augmentation vs. domain adaptation—A case study in human activity recognition. *Technologies* **2020**, *8*, 55. [CrossRef]
53. Vernikos, I.; Mathe, E.; Spyrou, E.; Mitsou, A.; Giannakopoulos, T.; Mylonas, P. Fusing handcrafted and contextual features for human activity recognition. In Proceedings of the 2019 14th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), Larnaca, Cyprus, 9–10 June 2019; pp. 1–6.
54. Liu, C.; Hu, Y.; Li, Y.; Song, S.; Liu, J. Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. *arXiv* **2017**, arXiv:1703.07475.
55. Lawton, M.P.; Brody, E.M. Assessment of older people: Self-maintaining and instrumental activities of daily living. *Gerontologist* **1969**, *9*, 179–186. [CrossRef]
56. Shahroudy, A.; Liu, J.; Ng, T.T.; Wang, G. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
57. Hu, J.F.; Zheng, W.S.; Lai, J.; Zhang, J. Jointly learning heterogeneous features for RGB-D activity recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5344–5352.
58. Xia, L.; Chen, C.C.; Aggarwal, J.K. View invariant human action recognition using histograms of 3d joints. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 20–27.
59. Aggarwal, A.; Mittal, M.; Battineni, G. Generative adversarial network: An overview of theory and applications. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100004. [CrossRef]
60. Jovanovic, M.; Campbell, M. Generative artificial intelligence: Trends and prospects. *Computer* **2022**, *55*, 107–112. [CrossRef]
61. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
62. Chollet, F. Keras. 2015. Available online: https://github.com/fchollet/keras (accessed on 3 March 2025).
63. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for Large-Scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
64. Vernikos, I.; Spyropoulos, T.; Spyrou, E.; Mylonas, P. Human Activity Recognition in the Presence of Occlusion. *Sensors* **2023**, *23*, 4899. [CrossRef]
65. Kwon, T.; Tekin, B.; Tang, S.; Pollefeys, M. Context-aware sequence alignment using 4D skeletal augmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8172–8182.
66. Li, B.; Cui, W.; Wang, W.; Zhang, L.; Chen, Z.; Wu, M. Two-stream convolution augmented transformer for human activity recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 19–21 May 2021; Volume 35, pp. 286–293.
67. Koutrintzes, D.; Mathe, E.; Spyrou, E. Boosting the Performance of Deep Approaches through Fusion with Handcrafted Features. In Proceedings of the ICPRAM, Online, 3–5 February 2022; pp. 370–377.

sensors

Article

Real-Time Tracking Data and Machine Learning Approaches for Mapping Pedestrian Walking Behavior: A Case Study at the University of Moratuwa

Harini Sawandi [1], Amila Jayasinghe [1] and Guenther Retscher [2,*]

[1] Department of Town & Country Planning, University of Moratuwa, Moratuwa 10400, Sri Lanka; harinisawandi@gmail.com (H.S.); amilabj@uom.lk (A.J.)
[2] Department of Geodesy and Geoinformation, TU Wien—Vienna University of Technology, 1040 Vienna, Austria
* Correspondence: guenther.retscher@tuwien.ac.at

Abstract: The growing urban population and traffic congestion underline the importance of building pedestrian-friendly environments to encourage walking as a preferred mode of transportation. However, a major challenge remains, which is the absence of such pedestrian-friendly walking environments. Identifying locations and routes with high pedestrian concentration is critical for improving pedestrian-friendly walking environments. This paper presents a quantitative method to map pedestrian walking behavior by utilizing real-time data from mobile phone sensors, focusing on the University of Moratuwa, Sri Lanka, as a case study. This holistic method integrates new urban data, such as location-based service (LBS) positioning data, and data clustering with unsupervised machine learning techniques. This study focused on the following three criteria for quantifying walking behavior: walking speed, walking time, and walking direction inside the experimental research context. A novel signal processing method has been used to evaluate speed signals, resulting in the identification of 622 speed clusters using K-means clustering techniques during specific morning and evening hours. This project uses mobile GPS signals and machine learning algorithms to track and classify pedestrian walking activity in crucial sites and routes, potentially improving urban walking through mapping.

Keywords: walking behavior; mobile GPS tracking; machine learning; pedestrian-friendly environment

Citation: Sawandi, H.; Jayasinghe, A.; Retscher, G. Real-Time Tracking Data and Machine Learning Approaches for Mapping Pedestrian Walking Behavior: A Case Study at the University of Moratuwa. *Sensors* **2024**, *24*, 3822. https://doi.org/10.3390/s24123822

Academic Editor: Valentina Agostini

Received: 8 May 2024
Revised: 4 June 2024
Accepted: 7 June 2024
Published: 13 June 2024

1. Introduction

Investigating pedestrian behavior and improving walking space in streets are becoming increasingly crucial considering the proven benefits to health, sustainability, and the development of safer pedestrian-friendly areas [1,2]. Consequently, an increasing amount of research has been carried out examining the relationship between the urban environment and individuals' behavior on streets [3]. However, many of these studies focus on the macro level; in addition to considering the urban characteristics on a wider scale, it is important to also consider microscale factors of urban design that influence behavior on streets [1]. This requires collecting data on the micro-level walking behavior of individuals on the streets to obtain precise information and develop target solutions for promoting walking.

Behavior mapping is a commonly utilized technique for the direct and systematic monitoring of individual behaviors and locations [1]. This mapping was first used in indoor locations, primarily in the fields of psychology, sociology, and criminology. It has since become commonly employed in public spaces like streets, parks, and playgrounds [4,5]. Currently, this mapping extends to street design and street planning. Shoval et al. [6] utilized psychological mapping to generate real-time maps of subjective and objective emotions to analyze Jerusalem's urban surroundings for the first time. Building upon that study, there is currently a growing interest in mapping behavior across several fields [7–9].

Recent technological advances have enabled these studies to use real-time surveying, tracking technologies, and global positioning systems (GPS). GPS data have been gathered since the 1990s to analyze transportation and access system performance, including measuring traffic flow, studying travel patterns, calculating route choices, etc. [10]. This opens a new research area by combining modern technical methods with real-time interactive communication. The availability of technology for the establishment of the geographic coordinates of mobile phones and other devices has significantly increased, leading to the emergence of a wide range of applications of location-based services (LBS) [11]. Similarly, Wi-Fi signals were also utilized in these studies [9]. The incorporation of LBS improves the quantitative metrics, offering significant spatial observation of the pattern of pedestrian movement. Smartphones, with their wireless connection, inertial sensors, and cameras, have greatly impacted digital health and gait analysis studies. Smartphones are currently providing physiological assessment and data entry/collection capabilities through several sensing modalities, which are accessible via apps [12].

However, this study found a few significant research gaps. The emergence of walking behavior depends on a combination of sociocultural variables, individual choices, and habitual patterns rather than a physical setting [13]. Therefore, to enhance our understanding of walking behavior patterns, we must acknowledge the complexity of this field.

Obtaining a detailed comprehension of individual experience in terms of time (second) and space (meters) offers new possibilities for study and strategic decision-making [6]. It is crucial to integrate the environment and individual characteristics of multiple individual inputs on the map to form a clear visual representation of different patterns [7]. The use of standard GPS positions is collected at intervals of a few seconds, resulting in hundreds of data points at each interval and large datasets for a thorough analysis. Thus, the first need is to map patterns of pedestrian walking behavior, as existing research highlights the importance of modern data gathering and analysis approaches, including objective walking patterns from large-scale tracking and machine learning analysis, to study the types of pedestrian walking patterns [14].

Most walking behavior studies have measured pedestrian walking utilizing walking duration, flows, and number of walkers [15]. Nevertheless, these studies are limited by their dependence on observational methodologies. Thus, while they offer new insights, their method has limitations [16]. Multiple authors identify the following five essential components in the observation process: 1. a visual representation of the observed areas; 2. a precise explanation of the human behavior observed, tracked, described, or outlined; 3. a timetable of recurring intervals for observation and recording; 4. a methodological observation process; and 5. a system for programming analysis that reduces the recording workload [1]. While walkability studies highlight the complexity of walking activities, there is still a lack of systematic categorization of pedestrian actions, creating a knowledge gap [9]. Further, research on walking patterns disregards the individual travel direction and time of the day. The individual travel direction has a significant impact on walking behavior changes; in addition, the direction in which people walk is influenced by built environment scenarios [17]. Also, time series analysis can be used to study changes in behavior over time [8]. The absence of this information presents difficulty in acquiring walking patterns. Hence, the second need involves considering individual travel direction and temporal factors to map and analyze walking behavior using real-time tracking applications. This approach allows researchers to collect more detailed and accurate data, resulting in more effective urban planning efforts.

The behavior of human walking is naturally complex and shaped by a variety of environmental and psychological factors. Subjective verification, which is frequently based on personal user experience and perception, plays a crucial role in validating the accuracy of measurement models, ensuring that they accurately reflect the complex nature of real-world walking scenarios. Current research on evaluating human walking behavior primarily concentrates on the construction of index systems, as well as data collecting and processing.

However, it does not include any verification of the reliability of measurement findings or the validity of measurement models [18].

In this context, analyzing pedestrian walking patterns using machine learning algorithms is critical for improving road safety and optimizing traffic management. Traditional techniques, such as artificial neural networks (ANNs) and hidden Markov models (HMMs), have been widely used in this domain [19]. However, these algorithms frequently encounter significant issues, such as overfitting and substantial data dependency, in real-time processing scenarios. For example, Ajmaya and Eklund's [19] study on recognizing pedestrian events using IMU and GPS data emphasized the challenges of improving ANN models, as well as the vast amount of training data required to obtain reliable results. Furthermore, the study by Gong et al. [20] identified limitations in using density-based spatial clustering of applications with noise (DBSCAN-TE) and support vector machine (SVM) methodologies, such as the need for parameter estimation and the lack of instant speed and acceleration features, which reduce the accuracy and efficiency of detecting pedestrian stops and movements.

Although data-driven methods have the potential to assist in making informed design decisions, it is still uncertain which new sources of information and approaches could be utilized to obtain insights into studying pedestrian walking behaviors in urban areas, resulting in a shortage of knowledge. Given the focus of our work, machine learning approaches have also been utilized in various areas of GNSS positioning. For instance, Zhang et al. [21] used unsupervised machine learning techniques to enhance precise positioning and navigation in complex environments. By integrating best integer equivariant (BIE) estimation with unsupervised K-means clustering algorithms, the proposed methods significantly improved both accuracy and reliability. The experiment demonstrates that the use of this approach could achieve millimeter-level precision, highlighting the effectiveness of machine learning techniques in such applications. In this study [21], K-means achieved high accuracy rates for inferring transportation modes, particularly when speed profiles were used as attributes. This implies that K-means clustering can handle the spatial and temporal patterns in pedestrian movement data without considerable parameter modification.

Given the limitations and potentialities in current studies, this research developed a framework utilizing unsupervised machine learning techniques to map pedestrian walking behavior in streets with real-time tracking data using mobile phones. This research endeavors to explore pedestrian walking behavior by utilizing mapping techniques to provide insight that exceeds traditional methods. Researchers can collect complex characteristics of pedestrian behavior, such as movement patterns, and interaction with the built environment using real-time tracking technologies and advanced mapping approaches. Integrating tracking technologies with subjective and objective experiences of human behavior could greatly enhance urban planning [6]. Extensive mapping not only helps to improve urban design to promote pedestrian activity but also creates convenient urban environments.

2. Materials and Methods

2.1. Case Study

We developed an approach for mapping pedestrian walking behavior on the streets of the University of Moratuwa and its vicinity. Campus walking areas are prioritized in urban sustainable development and developing pedestrian-friendly surroundings [2]. Given the nature of the experiment, this study examined walking behavior at the University of Moratuwa and its surrounds, as shown in Figure 1. The case study includes public locations, residential neighborhoods, cultural and commercial areas, and diverse university spaces to evaluate walking behavior patterns in varied environments and situations. Pedestrian traffic was seen in these areas, as many individuals were walking around the university because university students choose different routes daily to get to the university.

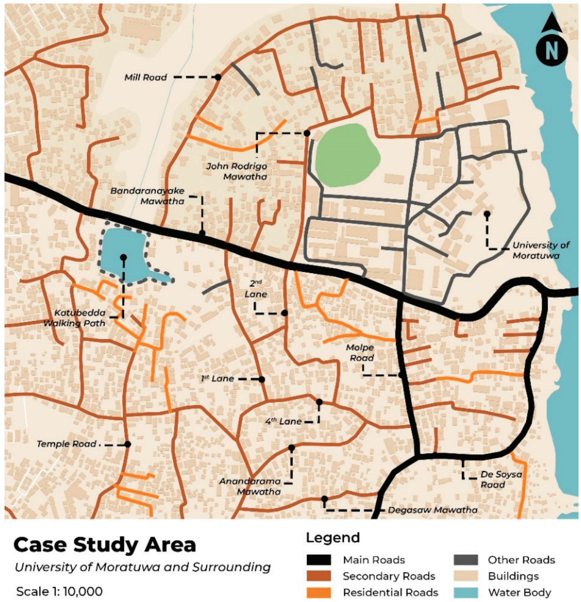


Figure 1. Case study area.

2.2. Experimental Design and Workflow

University students who live near the University of Moratuwa were selected as a sample. These students, aged 23 to 27 years, were chosen for this study because they were well-acquainted with their surroundings due to their daily commuting from their accommodation (boarding) to the university. Initially, 60 individual samples with a 50% gender distribution were chosen. The experiment was only carried out on weekdays. Weekday tests allowed for more focused monitoring of university pedestrian behavior. Specific times for the experiments were set to prevent overcrowding and congestion. The times were 7:00 to 9:00 a.m. and 4:00 to 6:00 p.m. These times of the day were carefully chosen to include both morning and evening peak hours to thoroughly examine pedestrian behavior.

As pointed out in the study by [18,22], which investigated several accelerometer placements, including at the hip (belt), wrist, upper arm, ankle, and thigh of the test person, using numerous accelerometers aids in activity identification. Thus, as illustrated in Figure 2, in this study, participants were advised to secure their mobile phones to their waistbands with designated holders. This strategy guaranteed that the phones remained in a consistent and steady position throughout the data collection process. The test persons were advised to keep the mobile phone in a vertical orientation within the holder to preserve consistency. This placement of the phones enabled the accurate gathering of data on walking speed, acceleration, and GPS position during the experiments.

Also, to effectively capture speed data when walking, the following parameters needed to be considered: holding the phone upright in portrait orientation by ensuring that the X-axis (horizontal) is parallel to the direction of movement while the Y-axis (vertical) is perpendicular to the ground. This orientation enables the phone's sensors to better capture forward movement (along the X-axis) and up-and-down motion (along the Y-axis), both of which are important for estimating speed.

Data for this study were collected using the Redmi Note 12 Pro smartphone, which has a powerful sensor suite that includes an accelerometer, gyroscope, and GPS receiver. Throughout the experiment, the use of these sensors was of utmost importance for collecting data on walking speed, acceleration, and geographic positions. Furthermore, the smartphone's long-lasting battery allowed for continuous data collection over lengthy periods.

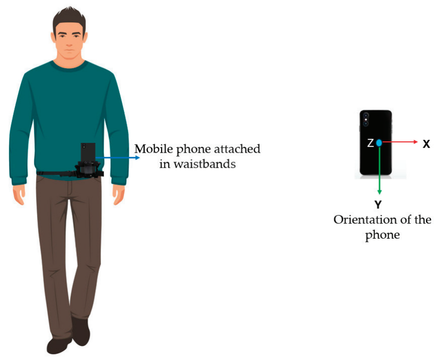


Figure 2. Experiment setup.

Each participant was instructed to use their mobile phone to record their walking speed, acceleration, and other variables available through the mobile app. Over four weeks, participants were required to record these characteristics as they traveled around the case study area during both the morning and evening. To minimize bias from long-distance walking and associated fatigue, each recording session was limited to 10 min. Accurate instructions on how to carry the phones were provided to participants. To ensure precise data collection, participants were additionally directed not to check their phones while walking and to keep focused on their surroundings. Every participant was informed of this study's purpose and the possible results, and they were all given specific instructions to explore and experience the environment throughout the study walking sessions. Throughout the process of this research, ethical standards and privacy have been carefully upheld.

2.3. Data Analysis

Clustering individuals based on their walking speed, direction, and time enables the precise classification of walking behavior. K-means clustering aids in identifying clusters [9]. Mapping identifies locations with high levels of pedestrian traffic and their distribution. Clustering identifies areas with high levels of foot traffic or congestion hotspots, allowing for more analysis to guide more effective solutions. Data analysis included three steps.

Framework of This Study

Figure 3 illustrates the framework used for evaluating location data and analyzing pedestrian walking behavior. This framework offers a structured approach to gaining insights from the dataset.

- Data collection using the sensor logger app;
- Data preparation—Before beginning the data preprocessing, each CSV file is classified based on the time of travel and travel direction to gain additional insights for a comprehensive dataset;
- Data preprocessing—The initial phase of the farmwork involves manual data preprocessing;
- Data preparation for K-means clustering—The dataset was cleaned and normalized to detect clusters of pedestrian walking behavior. We used a bespoke algorithm to preprocess the dataset;
- Mapping the results—The work principally centers on cluster analysis, employing unsupervised machine learning methods to reveal noteworthy trends and identify the homogeneous profile among pedestrians;
- Data validation—Data validation is conducted through the outputs of mapping using K-means clustering and expert subjective assessment.

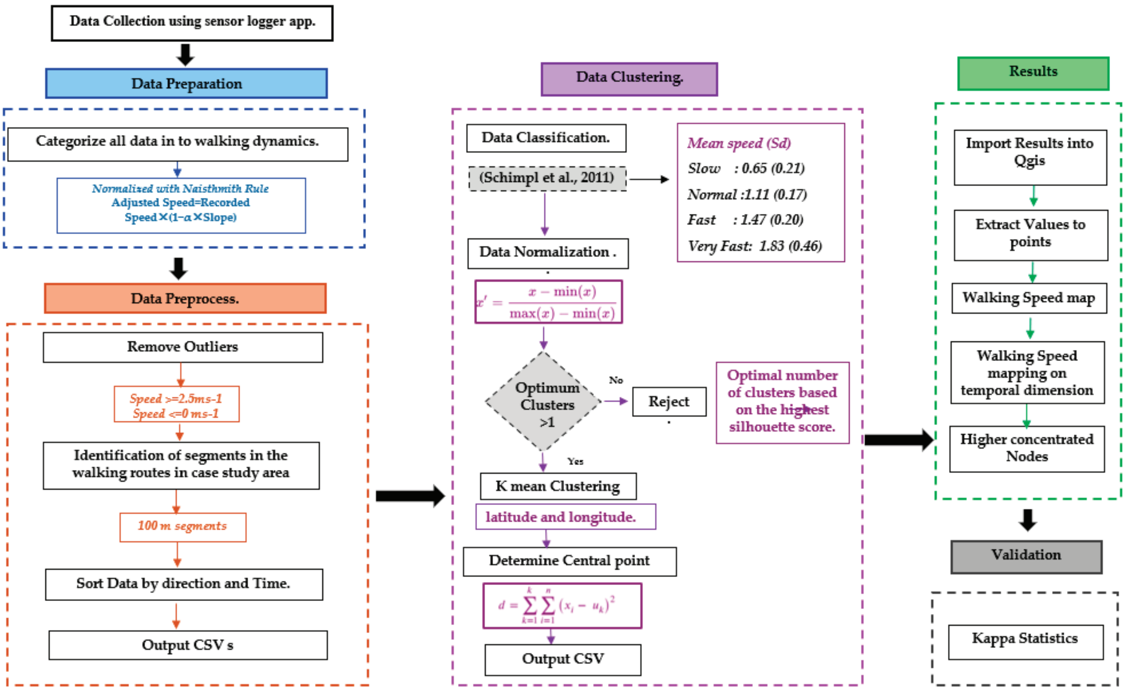


Figure 3. Framework of this study. Framework of this study. The data classification component under the data clustering is based on the methodology proposed by Schimpl et al. [23].

1. Data collection using sensor logger app.

Sensor logger is a smartphone app that collects objective participant data. Figure 4 illustrates the interface of the sensor logger mobile application, which is used to gather data. The application is available for Android, iPhone, and Apple Watch. The main reason for choosing this mobile app is its ability to adjust walking speed according to the Naisthmith rule, which sets it apart from other apps. It was useful for tracking small speeds caused by terrain changes when walking. Also, by using this mobile app, one can capture diverse walking dynamics. It mostly records the participant's geographical location and timestamp as they walk along the path. The program records acceleration, location, gyroscope, speed, step count, sound, heart rate, wrist motion, and other elements to capture the walking behavior. While the sensor logger app collects a range of data types, this analysis prioritizes the use of metrics appropriate to this study's aims. We specifically focus on using location data in conjunction with metrics like walking speed, longitude, latitude, accuracy, time, etc. In this mobile application, speed is calculated by measuring the change in consecutive GPS coordinates over time. The application determines speed by recording latitude and longitude at each time point and calculating the distance walked throughout each interval. The app's privacy practices may involve the management of subsequent data. The data can be exported in several forms, such as Zip, CSV, JSON, and SQLite.

The entire dataset was initially normalized using the Naisthmith method to ensure consistency in the measurements of walking speed across various terrains. The Naisthmith rule is a common technique for calculating the actual working speed when facing slopes or uneven surfaces.

The Naismith rule can be described using the following:

$$\text{Equivalent distance} = \text{Horizontal distance} + (\text{Vertical distance} * \alpha) \tag{1}$$

The horizontal distance refers to the distance at which an object moves on a level surface, whereas the vertical distance indicates the rate of ascent or descent. The parameter α represents Naisthmith's number, a constant coefficient that quantifies the additional exertion needed as a result of variations in altitude, which is usually set to 7.92.

Figure 4. Sensor logger mobile application.

2. Data Preparation

The data collected includes multiple characteristics, such as time, elapsed seconds, longitude, latitude, altitude, speed, bearing accuracy, vertical accuracy, horizontal accuracy, and bearing. The dataset chosen for this study is presented in Table 1. The mobile data covers 4 weeks and includes a complete dataset of 96,924 points. Before starting the data preprocessing, each CSV file based on time travel and travel direction has to be classified to gain additional insights for a comprehensive dataset.

Table 1. Types of data collected for this study.

Second Elapsed (Seconds)	Bearing Accuracy (Degrees)	Speed Accuracy (ms^{-1})	Vertical Accuracy (m)	Horizontal Accuracy (m)	Speed (ms^{-1})	Bearing (Degree)	Altitude (m)	Longitude (Degrees)	Latitude (Degrees)
4188.1	0	0.0806	34.74	9.94	0.0001	0	−77.2	79.9002	6.7956
2096.2	0	0	4.18	11.70	0.0001	0	−76.4	79.8990	6.7953
1699.1	0	0.15	1.13	13.32	0.0002	0	−73.8	79.9000	6.7963

3. Data preprocessing

The dataset included individual speed points captured using speed-tracking techniques in the mobile phone app. Following the framework proposed, an initial phase in the data preprocessing was the removal of outliers from each user's output. Outliers are samples that appear to be inconsistent with the overall trend of the GPS signal. They could be peaks, discontinuities, saturation, etc. To properly assess a signal, it must be removed without affecting the rest of the data. In the context of analyzing walking behavior patterns, outliers were defined as speed values outside the normal walking range, as follows: a speed of 0.0 ms^{-1}, indicating negligible movement, such as waiting, and a speed of 2.5 ms^{-1} or more, which is likely inaccurate due to GPS signal errors.

In addition to outlier removal, the dataset was analyzed for direction of travel and time. The manual process involved analyzing each user's outputs for GPS points and assigning the relevant direction based on timestamps using QGIS.

When assessing GPS signals for walking behavior, it is critical to focus on potential outliers induced by rapid and major shifts in the signal, such as significant braking. Before running the algorithm, the outlier removal strategy was used to thoroughly evaluate the data for outliers and ensure the correct classification of valid samples.

4. Data preparation for K-means clustering

Clustering techniques are divided into types based on splitting, density, and model. The K-means algorithm offers several advantages over other established approaches, such as straightforward mathematical principles, quick convergence, the improved scalability to big datasets, the effective management of high-dimensional datasets, and straightforward implementation. This approach is adaptable and can be utilized across several fields, as well as simply adapted to new scenarios. The reason for choosing K-means clustering is its capability to cluster data by reducing the sum of squared error (SSE) inside clusters.The sum of squared error (SSE) is given by the following equation:

$$d = \sum_{k=1}^{k} \sum_{i=1}^{n} (x_i - u_k)^2 \tag{2}$$

The main function of the sum of the squared error is represented by d, where k is the number of clusters, n is the number of observations, x_i is an observation i, and u_k is the centroid generated for the cluster of x_i.

However, typical K-means clustering has limitations. To address these issues in this unsupervised machine learning model, a framework has been developed by following specific techniques.

- Utilizes only numerical input variables—K-means uses distance-based metrics to analyze the similarity between data points, restricting the evaluation to only numerical factors. The analysis utilized geographic longitude and latitude coordinates to identify the pattern of walking behavior. In addition, the undefined (NaN) values were removed. Clustering results may be distorted if NaN values are included in the raw dataset (CSV output);
- Outlier removal in data classification—To cluster the data for studying walking behavior, the major dynamic being considered here is walking speed, which was collected through a mobile application. To categorize the speed of data, the existing literature has been examined. The speed property divides walking speeds into the following four categories: "Slow", "Normal", "Fast", and "Very Fast" [23]. It is imperative to evaluate the potential impact of outlier data on the K-means clustering analysis during the preparatory phase at this stage. The IQR-based outlier removal method was used on each speed category to remove data points that were outside the permitted range. Table 2 shows the average value of accuracy in each cluster after removing the outliers and categorizing them into clusters.

Table 2. Average values of each cluster used in K-means clustering.

Cluster	Bearing Accuracy (Degrees)	Speed Accuracy (ms^{-1})	Vertical Accuracy (m)	Horizontal Accuracy (m)	Speed (ms^{-1})
Slow	0	0	0.259	0.543	0.6775
Normal	0	0.1485	0	0.600	1.1081
Fast	0	0	0	0.677	1.4091
Very Fast	0	0	0.548	0.667	1.5447

- Data Normalization—Data normalization was performed using the min–max scaler method [24] in Python using the sci-kit-learn package. The min–max scaler is given as follows:

  $$x^1 = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{3}$$

  Let x^1 represent the normalized value, x represents the initial value within a particular range, $\min(x)$ represents the minimum value of the attribute within that range, and $\max(x)$ represents the maximum attribute value within that range.
  This phase ensured that the results were not affected by variations in scales and that all scales had an equal impact on model fitting.
- The optimal number of clusters—This study utilized the K-means method to identify unique patterns in the data based on geographical coordinates (latitude and longitude)

and walking speeds. Clustering algorithms depend on a random initialization of the cluster centroid. Silhouette analysis (SA) was used to address the issue and determine the ideal number of clusters for each speed category [25]. Introduced by Rousseeuw in 1987, the silhouette analysis (SA) technique calculates the silhouette score, a statistic that varies between −1 and 1. This score provides information on the proximity and density of clusters, indicating their closeness or distance from one other and the total density of the clusters.

5. Mapping the Results

The silhouette analysis (SA) approach helped to identify the ideal number of clusters within each speed category. The next step was to map these clusters to reveal distinctive patterns within the case study area. Basic QGIS mapping techniques were used to accomplish this.

6. Data Validation

Validating the results is essential after finishing the analytical process. Validating these results is challenging because spatial quality is a normative criterion [26]. This study used a unique method that contrasted the results of the unsupervised machine learning algorithm with the preferred speed of each cluster as identified by the students participating in the research.

For this method, 150 clusters, which is one-third of all cluster setups, were carefully chosen for assessment. These clusters were specifically used because they were within one standard deviation. A total of 150 pictures were manually captured during the evaluation procedure in the study area. Travel direction and traveled time were considered for each cluster when taking the photos. The research participants were required to walk in a specific direction and only identify what was directly in front of them; thus, the analysis disregards the whole 360-degree field of vision and instead focuses on a 120-degree field of view based on their walking path.

The camera features an 8-megapixel resolution and an f/2.2 aperture, ensuring satisfactory image detail and effective light capture, respectively. Featuring a 120-degree ultrawide field of vision, this device is capable of effectively capturing wide and expansive scenes as well as group photographs. Furthermore, the camera's 1/4.0-inch sensor size and 1.12 µm pixel size is advantageous. Students were asked to rate their preferred speed of walking when looking at images categorized as slow, normal, fast, and very quick.

3. Results

3.1. Results of Cluster Mapping

By using the K-means clustering techniques, a total of 622 speed clusters were identified during both morning hours (07.00 to 09.00 a.m.) and evening hours (4.00 to 06.00 p.m.) independently. The results discussed below are based on collected data and represent the areas most extensively used by students. GIS and spatial analysis are employed to map the results of the clustering of walking behavior, which includes point distribution maps that illustrate pedestrian concentration on the road.

This study mapped pedestrian density in various locations within each of the four speed categories during morning and evening hours. Figure 5 illustrates the spatial distribution of pedestrian density among different speed clusters in the morning, whereas Figure 6 illustrates pedestrian concentration in the evening. The identification of unique spatial behaviors within each cluster is facilitated by the display of data in both morning and evening timestamps, which enables the observation of a variety of patterns. Figures 7 and 8 depict the walking behavior patterns in the morning and evening based on cluster analysis. The points indicate the centroid of each cluster derived from K-means clustering.

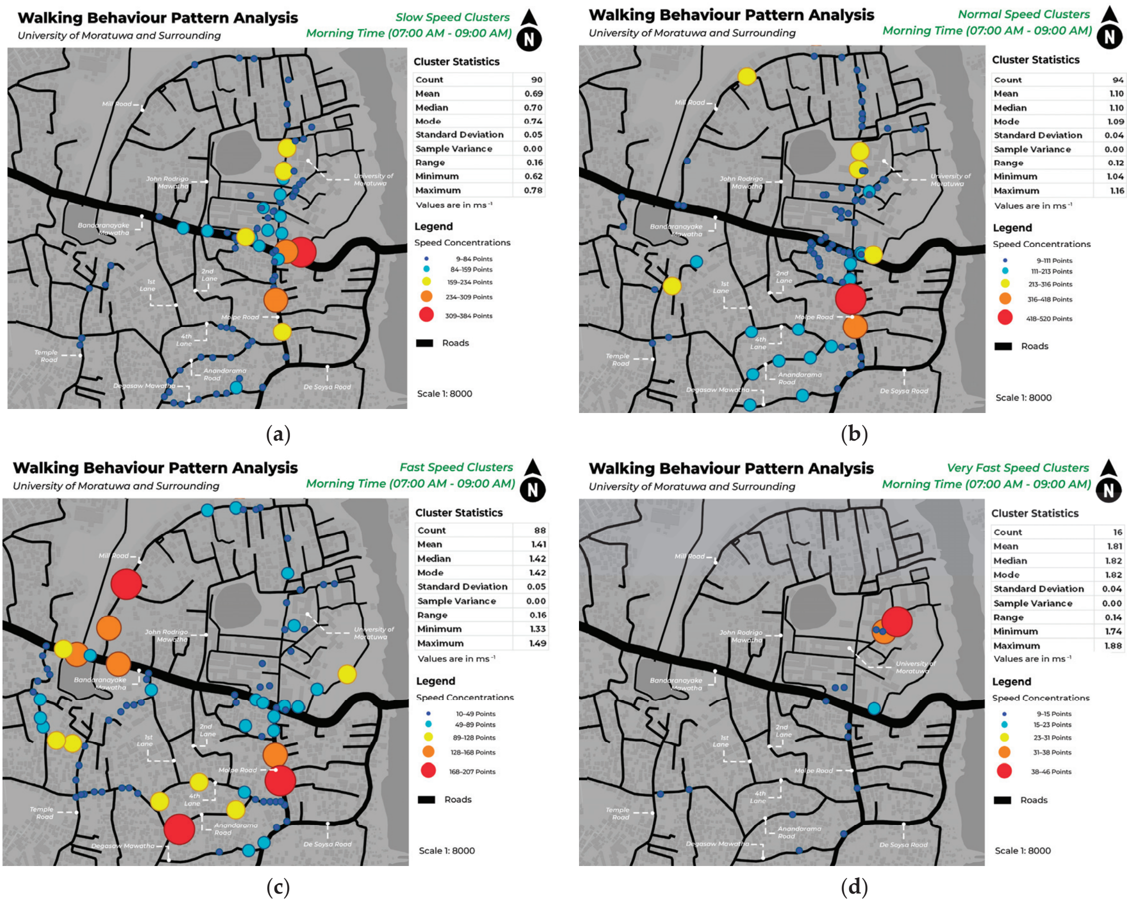


Figure 5. Pedestrian concentration in slow walking speeds (**a**), normal walking speeds (**b**), fast walking speeds (**c**), and very fast walking speeds (**d**) in the morning hours.

3.2. Results of Data Validation

Appendix A shows the results of ratings obtained by machine learning and the recordings from the students. For each of the 150 clusters, two ratings were estimated. Out of the 150 ratings, 126 were found to be the same, which is considered acceptable. The results were assessed for validity using Kappa Statistics to quantify the inter-rater reliability between the ratings made using machine learning and the recordings from the students. The values were calculated using Python. Table 3 presents the summary of the results.

Table 3. Statistics of the machine learning and the recordings from the students indicating the agreement value.

Agreement	Cohen's Kappa Coefficient	Std. Err.
85.3%	0.8	0.013

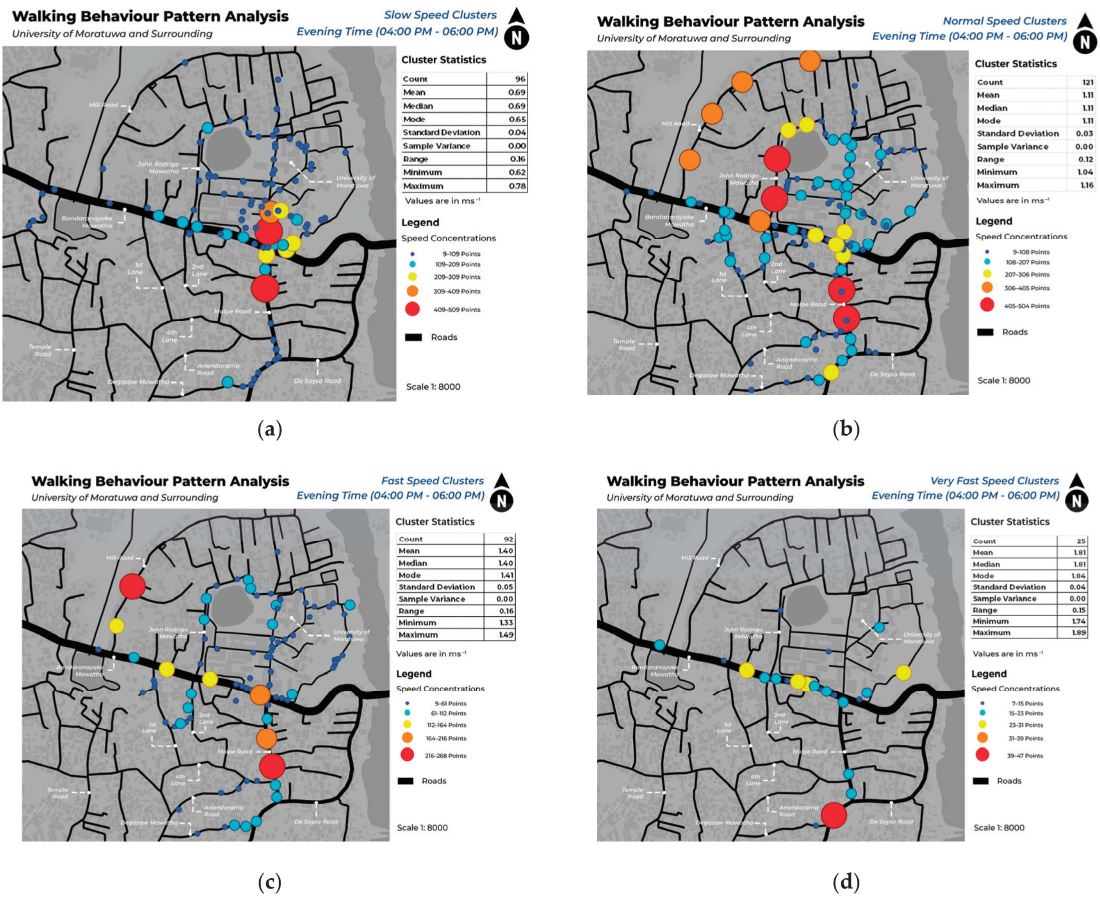


Figure 6. Pedestrian concentration in slow walking speeds (**a**), normal walking speeds (**b**), fast walking speeds (**c**), and very fast walking speeds (**d**) in the evening hours.

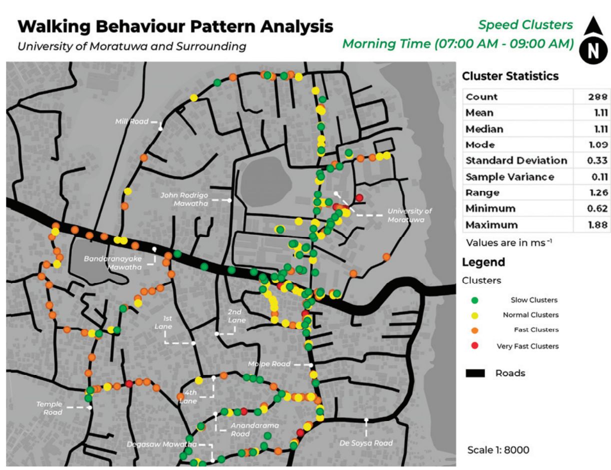


Figure 7. Pedestrian concentration in different walking speed ranges in the morning hours.

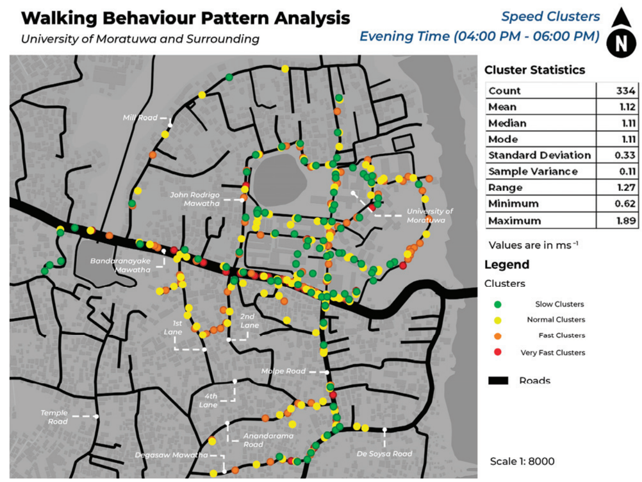


Figure 8. Pedestrian concentration in different walking speed ranges in the evening hours.

4. Discussion

This research enhances the understanding of the complex connections between urban environments and pedestrian behavior by utilizing quantitative location data and speed data of pedestrians, along with qualitative validation of the findings. "Big Data" analysis is used to process and analyze significant and complicated datasets that typical data processing systems are not able to handle. Various methods exist for gathering and analyzing vast amounts of data, but there is a need for a standardized framework to extract insights. In this paper, a method is proposed for conducting experiments using the pedestrian to collect and analyze data and extract insights to conclude behavior, rather than relying on biased observational data or the existing literature.

This study's findings provide useful insights into pedestrian behavior patterns within various speed categories. The clustering study using unsupervised machine learning discovered several clusters for each of the following speed categories: slow, normal, fast, and very fast. The clusters indicate regions with different pedestrian densities and speeds of movement. Mapping highlights locations with a significant concentration of each cluster on different roadways. The slow mean walking speed is 0.69 ms^{-1}, the normal mean walking speed is 1.11 ms^{-1}, the fast mean speed is 1.41 ms^{-1}, and the very fast mean speed is 1.81 ms^{-1}.

In this case, both in the morning and evening hours, a significant increase in pedestrian traffic on Bandaranayake Mawatha and Molpe Road can be observed. In those areas, there is a high concentration of slow and normal walking pedestrians. The field observations show that these locations correspond to the main entry for vehicles at the University of Moratuwa, where there is a notable rise in vehicular traffic and dense retail structures along Molpe Road.

According to a study by [27], young adults walk at a speed of 89 m/min in educational areas and 80 m/min in commercial areas. According to the speed ranges in this study, the speed falls within the normal range. Results show that pedestrians passing around the university area (Bandaranayake Mawatha) and the nearby commercial area (Molpe Road) tend to walk at a normal speed, which aligns with the existing findings of the aforementioned literature.

Inside the University of Moratuwa, during the morning hours, there is a high degree of pedestrian concentration observed in the Lagan area. This congestion can be categorized as slow and normal speed clusters. Based on on-site observation, this location in the university is predominantly a green environment, which is similar to findings of [28] research that indicates that individuals tend to walk slower in greener environments. In addition, the concentration of speed in different areas of the University of Moratuwa is constant.

This study found a substantial correlation (85%) between machine-generated scores and subjective assessments. This verification demonstrates the effectiveness of our strategy, which benefits substantially from the large dataset employed for analysis.

Finally, using the experimental study, we were able to visualize, accurately analyze, and validate walking behavior patterns. Visualization is essential for managing tracking data. GPS is important in urban research because it provides precise and observable data that can be combined to create a new evidence base for predicting future urban trends.

5. Conclusions and Outlook

The objective of this research is to present an approach for analyzing pedestrian paths in urban environments. Individuals' walking patterns are evaluated on multiple streets inside the case study location. Our employed method provides a full understanding of pedestrian dynamics by gathering real-time tracking data and using unsupervised machine learning methods to assess walking behavior.

This work provides insights into walking behaviors. Firstly, this study adds to the current body of knowledge by developing a theoretical basis for using unsupervised K-means clustering machine learning algorithms to assess pedestrian walking behavior using large-scale GPS data. Secondly, our work expands the transportation planning and urban design and planning literature by using large-scale data analysis with established methodologies.

The authors' analysis of outcomes utilizing machine learning algorithms and mobile location data collection led to the following conclusions and interpretations. K-means clustering was used to determine the number of unique clusters within the case study area. This method was tested at various times of day and discovered that there is a different pattern in morning and evening walking behavior, demonstrating constancy in pedestrian concentration in the studied area. These patterns may arise as a result of urban design components and activities in an area that influences pedestrian movement patterns. Therefore, it is recommended in future research to analyze the spatial and environmental aspects of the study region to enrich the existing body of knowledge.

This study, however, is limited by continuous data collection due to technical challenges, resulting in missing information on certain days. This study only collected pedestrian movement data for a limited number of days and had a small sample size comprising only university students. The primary objective of this work was to examine the suggested methodology and showcase its efficacy in real-time tracking data by employing machine learning techniques to analyze pedestrian locomotion patterns. This study was conducted across two time periods, focusing on temporal aspects. Future research could benefit from extending the duration of observations, analyzing the temporal patterns of different activities in depth, and studying the diverse street users.

Author Contributions: Conceptualization, A.J. and H.S.; methodology, H.S., A.J. and G.R.; software, H.S. and A.J.; validation, H.S., A.J. and G.R.; formal analysis, H.S.; investigation, H.S.; resources, A.J. and G.R.; data curation, H.S.; writing—original draft preparation, H.S.; writing—review and editing, A.J. and G.R.; visualization, H.S.; supervision, A.J.; project administration, A.J.; funding acquisition, A.J. and G.R. All authors have read and agreed to the published version of the manuscript.

Funding: Open Access Funding by TU Wien.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data supporting the reported results were generated during the data collection phase of this study. Due to privacy and ethical restrictions, the data are not publicly available.

Acknowledgments: The author acknowledges the funding for the mobile phones from project 618657-EPP-1-2020-1-AT-EPPKA2-CBHE-JP from the Erasmus+ Capacity Building in Higher Education program. This project has been funded with support from the European Commission. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use that may be made of the information contained therein.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Summary table of the evaluation using machine learning and subjective evaluation conducted by the students.

Cluster No	Machine Evaluation	Subjective Evaluation	Cluster No	Machine Evaluation	Subjective Evaluation
1	Very Fast	Very Fast	76	Fast	Fast
2	Very Fast	Very Fast	77	Normal	Normal
3	Very Fast	Fast	78	Fast	Fast
4	Very Fast	Fast	79	Fast	Fast
5	Fast	Fast	80	Fast	Fast
6	Very Fast	Very Fast	81	Normal	Fast
7	Very Fast	Very Fast	82	Very Fast	Fast
8	Normal	Normal	83	Fast	Normal
9	Very Fast	Fast	84	Slow	Slow
10	Very Fast	Very Fast	85	Normal	Normal
11	Very Fast	Very Fast	86	Fast	Fast
12	Slow	Normal	87	Very Fast	Very Fast
13	Slow	Slow	88	Very Fast	Fast
14	Fast	Fast	89	Fast	Fast
15	Slow	Slow	90	Very Fast	Very Fast
16	Very Fast	Very Fast	91	Fast	Fast
17	Normal	Normal	92	Slow	Slow
18	Normal	Normal	93	Normal	Fast
19	Fast	Fast	94	Fast	Fast
20	Fast	Very Fast	95	Very Fast	Very Fast
21	Very Fast	Very Fast	96	Slow	Slow
22	Fast	Fast	97	Fast	Fast
23	Normal	Normal	98	Normal	Normal
24	Fast	Fast	99	Slow	Slow
25	Slow	Slow	100	Slow	Slow
26	Fast	Fast	101	Normal	Fast
27	Slow	Slow	102	Normal	Normal
28	Slow	Slow	103	Normal	Normal
29	Fast	Fast	104	Fast	Fast
30	Normal	Normal	105	Slow	Slow
31	Normal	Normal	106	Normal	Normal
32	Normal	Normal	107	Normal	Normal
33	Normal	Normal	108	Slow	Slow
34	Normal	Normal	109	Very Fast	Very Fast
35	Normal	Normal	110	Slow	Slow
36	Very Fast	Very Fast	111	Very Fast	Very Fast
37	Very Fast	Fast	112	Fast	Fast
38	Fast	Fast	113	Fast	Fast
39	Fast	Normal	114	Fast	Fast
40	Slow	Slow	115	Fast	Fast
41	Slow	Slow	116	Slow	Slow
42	Very Fast	Very Fast	117	Slow	Normal
43	Slow	Slow	118	Fast	Fast
44	Very Fast	Very Fast	119	Slow	Slow
45	Slow	Slow	120	Normal	Slow

Table A1. *Cont.*

Cluster No	Machine Evaluation	Subjective Evaluation	Cluster No	Machine Evaluation	Subjective Evaluation
46	Normal	Normal	121	Normal	Normal
47	Slow	Slow	122	Normal	Normal
48	Fast	Fast	123	Slow	Slow
49	Normal	Normal	124	Slow	Slow
50	Slow	Normal	125	Slow	Slow
51	Normal	Normal	126	Normal	Normal
52	Slow	Slow	127	Slow	Normal
53	Very Fast	Very Fast	128	Normal	Normal
54	Normal	Normal	129	Normal	Normal
55	Slow	Slow	130	Very Fast	Normal
56	Very Fast	Very Fast	131	Slow	Fast
57	Slow	Slow	132	Very Fast	Very Fast
58	Normal	Normal	133	Very Fast	Very Fast
59	Normal	Normal	134	Slow	Slow
60	Fast	Fast	135	Slow	Slow
61	Slow	Slow	136	Normal	Normal
62	Normal	Normal	137	Fast	Very Fast
63	Normal	Slow	138	Normal	Normal
64	Slow	Slow	139	Fast	Very Fast
65	Slow	Slow	140	Slow	Slow
66	Slow	Slow	141	Very Fast	Very Fast
67	Very Fast	Very Fast	142	Slow	Slow
68	Normal	Normal	143	Very Fast	Very Fast
69	Very Fast	Very Fast	144	Very Fast	Very Fast
70	Normal	Normal	145	Normal	Normal
71	Normal	Fast	146	Normal	Normal
72	Slow	Slow	147	Normal	Normal
73	Very Fast	Very Fast	148	Normal	Normal
74	Fast	Fast	149	Fast	Normal
75	Very Fast	Very Fast	150	Fast	Fast

References

1. Parra-Ovalle, D.; Miralles-Guasch, C.; Marquet, O. Pedestrian street behavior mapping using unmanned aerial vehicles. A case study in Santiago de Chile. *PLoS ONE* **2023**, *18*, e0282024. [CrossRef] [PubMed]
2. Ma, Y.; Zhang, J.; Yang, X. Effects of Audio-Visual Environmental Factors on Emotion Perception of Campus Walking Spaces in Northeastern China. *Sustainability* **2023**, *15*, 15105. [CrossRef]
3. Mehta, V. Lively streets: Determining environmental characteristics to support social behavior. *J. Plan. Educ. Res.* **2007**, *27*, 165–187. [CrossRef]
4. Goličnik, B.M.; Thompson, C.W. Emerging relationships between design and use of urban park spaces. *Landsc. Urban Plan.* **2010**, *94*, 38–53. [CrossRef]
5. Cosco, N.G.; Moore, R.; Islam, M.Z. Behavior mapping: A method for linking preschool physical activity and outdoor design. *Med. Sci. Sports Exerc.* **2010**, *42*, 513–519. [CrossRef] [PubMed]
6. Shoval, N.; Schvimer, Y.; Tamir, M. Tracking technologies and urban analysis: Adding the emotional dimension. *Cities* **2018**, *72*, 34–42. [CrossRef]
7. Zhu, M.; Teng, R.; Wang, C.; Wang, Y.; He, J.; Yu, F. Key environmental factors affecting perceptions of security of night-time walking in neighbourhood streets: A discussion based on fear heat maps. *J. Transp. Health* **2023**, *32*, 101636. [CrossRef]
8. Xiang, L.; Cai, M.; Ren, C.; Ng, E. Modeling pedestrian emotion in high-density cities using visual exposure and machine learning: Tracking real-time physiology and psychology in Hong Kong. *Build. Environ.* **2021**, *205*, 108273. [CrossRef]
9. Stanitsa, A.; Hallett, S.H.; Jude, S. Investigating pedestrian behaviour in urban environments: A Wi-Fi tracking and machine learning approach. *Multimodal Transp.* **2023**, *2*, 100049. [CrossRef]
10. Murgano, E.; Caponetto, R.; Pappalardo, G.; Cafiso, S.D.; Severino, A. A novel acceleration signal processing procedure for cycling safety assessment. *Sensors* **2021**, *21*, 4183. [CrossRef]
11. Ratti, C.; Frenchman, D.; Pulselli, R.M.; Williams, S. Mobile landscapes: Using location data from cell phones for urban analysis. *Environ. Plan. B Plan. Des.* **2006**, *33*, 727–748. [CrossRef]
12. Young, F.; Mason, R.; Morris, R.E.; Stuart, S.; Godfrey, A. IoT-enabled gait assessment: The next step for habitual monitoring. *Sensors* **2023**, *23*, 4100. [CrossRef] [PubMed]

13. Miah, S.; Milonidis, E.; Kaparias, I.; Karcanias, N. An Innovative Multi-Sensor Fusion Algorithm to Enhance Positioning Accuracy of an Instrumented Bicycle. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1145–1153. [CrossRef]
14. Feng, Y.; Duives, D.; Daamen, W.; Hoogendoorn, S. Data collection methods for studying pedestrian behaviour: A systematic review. *Build. Environ.* **2021**, *187*, 107329. [CrossRef]
15. Silitonga, S. Walkability; The relationship of walking distance, walking time and walking speed. *J. Rekayasa Konstruksi Mekanika Sipil (JRKMS)* **2020**, *3*, 19–26. [CrossRef]
16. De Arruda Campos, M.B.; Chiaraida, A.; Smith, A.; Stonor, T.; Takamatsu, S. Towards a "walkability index". In Proceedings of the European Transport Conference (ETC), Strasbourg, France, 8–10 October 2003. Available online: https://trid.trb.org/view/771383 (accessed on 1 June 2024).
17. Nishio, T.; Niitsuma, M. Environmental map building to describe walking dynamics for determination of spatial feature of walking activity. In Proceedings of the 2019 IEEE 28th International Symposium on Industrial Electronics (ISIE), Vancouver, BC, Canada, 12–14 June 2019. [CrossRef]
18. Wan, T.; Lu, W.; Sun, P. Constructing the quality measurement model of street space and its application in the old town in Wuhan. *Front. Public Health* **2022**, *10*, 816317. [CrossRef] [PubMed]
19. Ahmed, M.U.; Brickman, S.; Dengg, A.; Fasth, N.; Mihajlovic, M.; Norman, J. A Machine Learning Approach to Classify Pedestrians' Events based on IMU and GPS. *Int. J. Artif. Intell.* **2019**, *17*, 154–167.
20. Gong, L.; Yamamoto, T.; Morikawa, T. Identification of activity stop locations in GPS trajectories by DBSCAN-TE method combined with support vector machines. *Transp. Res. Procedia* **2018**, *32*, 146–154. [CrossRef]
21. Zhang, Z.; Li, X.; Yuan, H. Best Integer Equivariant Estimation based on Unsupervised Machine Learning for GNSS Precise Positioning and Navigation in Complex Environments. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *60*, 2672–2682. [CrossRef]
22. Twomey, N.; Diethe, T.; Fafoutis, X.; Elsts, A.; McConville, R.; Flach, P.; Craddock, I. A comprehensive study of activity recognition using accelerometers. *Informatics* **2018**, *5*, 27. [CrossRef]
23. Schimpl, M.; Lederer, C.; Daumer, M. Development and Validation of a New Method to Measure Walking Speed in Free-Living Environments Using the Actibelt® Platform. *PLoS ONE* **2011**, *6*, e23080. [CrossRef] [PubMed]
24. Visalakshi, N.K.; Suguna, J. K-means clustering using Max-MIN distance measure. In Proceedings of the NAFIPS 2009—2009 Annual Meeting of the North American Fuzzy Information Processing Society, Cincinnati, OH, USA, 14–17 June 2009; IEEE: Piscataway, NJ, USA, 2009. [CrossRef]
25. Yuan, C.; Yang, H. Research on K-value selection method of K-means clustering algorithm. *J* **2019**, *2*, 226–235. [CrossRef]
26. Zhang, L.; Ye, Y.; Zeng, W.; Chiaradia, A. A systematic measurement of street quality through multi-sourced urban data: A human-oriented analysis. *Int. J. Environ. Res. Public Health* **2019**, *16*, 1782. [CrossRef] [PubMed]
27. Rastogi, R.; Thaniarasu, I.; Chandra, S. Design implications of walking speed for pedestrian facilities. *J. Transp. Eng.* **2011**, *137*, 687–696. [CrossRef]
28. Franěk, M.; Režný, L. Environmental features influence walking speed: The effect of urban greenery. *Land* **2021**, *10*, 459. [CrossRef]

sensors

Article

Shape Sensing and Kinematic Control of a Cable-Driven Continuum Robot Based on Stretchable Capacitive Sensors

Wenjun Shen [1,2], Jianhui He [1,2], Guilin Yang [1,2,*], Xiangjie Kong [1,2], Haotian Bai [1,2,3] and Zaojun Fang [1,2]

1 Zhejiang Key Laboratory of Robotics and Intelligent Manufacturing Equipment Technology, Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo 315201, China; shenwenjun@nimte.ac.cn (W.S.); hejianhui@nimte.ac.cn (J.H.); kongxiangjie@nimte.ac.cn (X.K.); baihaotian@nimte.ac.cn (H.B.); fangzaojun@nimte.ac.cn (Z.F.)

2 College of Materials Science and Opto-Electronic Technology, University of Chinese Academy of Sciences, Beijing 100049, China

3 Nottingham Ningbo China Beacons of Excellence Research and Innovation Institute, School of Computer Science, University of Nottingham, Ningbo 315199, China

* Correspondence: glyang@nimte.ac.cn

Abstract: A Cable-Driven Continuum Robot (CDCR) that consists of a set of identical Cable-Driven Continuum Joint Modules (CDCJMs) is proposed in this paper. The CDCJMs merely produce 2-DOF bending motions by controlling driving cable lengths. In each CDCJM, a pattern-based flexible backbone is employed as a passive compliant joint to generate 2-DOF bending deflections, which can be characterized by two joint variables, i.e., the bending direction angle and the bending angle. However, as the bending deflection is determined by not only the lengths of the driving cables but also the gravity and payload, it will be inaccurate to compute the two joint variables with its kinematic model. In this work, two stretchable capacitive sensors are employed to measure the bending shape of the flexible backbone so as to accurately determine the two joint variables. Compared with FBG-based and vision-based shape-sensing methods, the proposed method with stretchable capacitive sensors has the advantages of high sensitivity to the bending deflection of the backbone, ease of implementation, and cost effectiveness. The initial location of a stretchable sensor is generally defined by its two endpoint positions on the surface of the backbone without bending. A generic shape-sensing model, i.e., the relationship between the sensor reading and the two joint variables, is formulated based on the 2-DOF bending deflection of the backbone. To further improve the accuracy of the shape-sensing model, a calibration method is proposed to compensate for the location errors of stretchable sensors. Based on the calibrated shape-sensing model, a sliding-mode-based closed-loop control method is implemented for the CDCR. In order to verify the effectiveness of the proposed closed-loop control method, the trajectory tracking accuracy experiments of the CDCR are conducted based on a circle trajectory, in which the radius of the circle is 55 mm. The average tracking errors of the CDCR measured by the Qualisys motion capture system under the open-loop and the closed-loop control are 49.23 and 8.40 mm, respectively, which is reduced by 82.94%.

Keywords: cable-driven continuum robot; stretchable capacitive sensors; shape-sensing model; closed-loop control

Citation: Shen, W.; He, J.; Yang, G.; Kong, X.; Bai, H.; Fang, Z. Shape Sensing and Kinematic Control of a Cable-Driven Continuum Robot Based on Stretchable Capacitive Sensors. *Sensors* **2024**, *24*, 3385. https://doi.org/10.3390/s24113385

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 25 April 2024
Revised: 18 May 2024
Accepted: 18 May 2024
Published: 24 May 2024

1. Introduction

A Cable-Driven Continuum Robot (CDCR) is a multi-degree-of-freedom mechanism actuated by light cables, which consists of a number of identical Cable-Driven Continuum Joint Modules (CDCJMs) [1–3]. Each CDCJM is composed of a base platform, a pattern-based flexible backbone, four driving cables, and a moving platform. Supported by the flexible backbone, the 2-DOF bending motions of the CDCJM are realized by controlling the driving cable lengths. The CDCR has the advantages of high flexibility and good adaptability. Therefore, it has been widely applied for dexterous manipulation in confined and complex spaces, such as minimally invasive surgery [4–7], collapsed buildings [8], space station mock-up environments [9,10], and the on-wing inspection of gas turbine engines [11,12].

A CDCR generally adopts a flexible backbone as its passive compliant joint to realize the 2-DOF bending deflections. Due to the disadvantage of low stiffness, the bending deflection of the flexible backbone is determined by not only the driving cable lengths but also the gravity and payload, which results in an inaccurate kinematic model of the CDCR. Therefore, the shape sensing of the flexible backbone is significant for a CDCR, since employs external measurement devices to measure the bending shape of the flexible backbone.

The commonly used shape-sensing methods of the continuum robots include the electromagnetic (EM)-tracking-based shape-sensing method [13–15], the vision-based shape-sensing method [16–18], and the shape-sensing method based on Fiber Bragg Grating (FBG) sensor [19–21]. The EM-tracking-based method employs an EM tracking system to simultaneously realize tip tracking and the shape measurement of the continuum robots. The EM tracking system consists of the EM field generator and multiple distributed EM sensors, in which the EM sensors are located along the continuum robots. Based on the position and orientation of each EM sensor, shape reconstruction algorithms are proposed in conjunction with the kinematic models of continuum robots. In [22], an extended Kalman filter is employed for the shape estimation of a surgical snake robot based on the EM sensor data. In [23], the shape reconstruction algorithm based on the quadratic Bezier curves is conducted for a CDCR, in which the pose information measured through the EM sensors and the length information of the robot are utilized to fit the shape of the robot. The shape-sensing method based on EM sensors is easy to integrate due to the small size of each EM sensor. However, it will result in the distortion of the EM field when there exist magnetic and conductive objects, which will decrease the measurement accuracy of the EM tracking system. Furthermore, the EM tracking method has limited workspace, and its tracking accuracy varies with the distance from the center of the EM field.

The vision-based shape-sensing method employs stereo cameras, infrared cameras, or high-speed cameras to measure the shape of the CDCR. In [24], the shape-sensing method based on stereo vision is proposed, which employs the Self-Organizing Mapping (SOM) algorithm for the shape reconstruction of a continuum robot based on the point cloud derived by the cameras. In [25], a marker-based shape-sensing method is proposed, in which multiple markers are located along the continuum robot. The positions of markers are measured through cameras, which are utilized to calculate the actual shape of the continuum robot. Although the vision-based shape-sensing method has high measurement accuracy, the measurement results are significantly affected by the external environment, which limits its application in confined spaces. Furthermore, the vision-based shape-sensing method has a serious time delay, which is not conducive to real-time control of a CDCR.

The FBG-sensor-based shape-sensing method employs the changes in fiber wavelengths to estimate the shape of continuum robots. In [21], the FBG fiber is inserted into the continuum robot with 1-DOF bending motion, in which the strain of the FBF fiber is employed to calculate the curvature of the continuum robot. In [26], the helical FBG sensors are located on the continuum robot, which are utilized for the curvature, torsion, and force measurement. The FBG-sensor-based shape-sensing method has the advantages of high resolution, high sensitivity, and high signal-to-noise ratio. However, the FBG sensors have to employ the interrogation system to demodulate the wavelengths of fibers, which is not conducive to integration.

The shape-sensing methods mentioned above cannot simultaneously have the advantages of high measuring accuracy, strong anti-interference capability, and the ability to measure the large bending deflection of the flexible backbone in confined spaces. Therefore, the shape-sensing method based on stretchable capacitive sensors is employed in this work. The stretchable capacitive sensors are a class of elastic strain sensors, which are made of flexible fabric and conductive nanomaterial to realize strain detection. They have the advantages of a high stretch rate, high sensitivity, and high stability. The capacitive value of the sensor will increase with its stretched length. The conventional shape-sensing methods based on stretchable sensors employ the parallel sensor location scheme, i.e., the location directions of sensors are parallel to the axis direction of the flexible backbone [27]. Such a sensor location scheme usually adopts four stretchable sensors for the shape measurement of the backbone with the 2-DOF bending deflections, which has good robustness. When the backbone produces the bending deflections, the shapes of sensors are assumed as a circle attached to the surface of the backbone, which can simplify the shape-sensing modeling analysis. However, the accuracy of the shape-sensing model with such a sensor location scheme will significantly decrease when there are location position errors of stretchable sensors. Furthermore, the parallel location scheme will increase the number of sensors.

In this paper, shape-sensing and closed-loop control based on stretchable capacitive sensors is proposed for the CDCR with a pattern-based flexible backbone. For the CDCJM with 2-DOF bending motions, two stretchable capacitive sensors are employed to measure the bending deflection of the flexible backbone so as to calculate the joint variables of the CDCJM. Two endpoints of each sensor are located on the surface of the backbone, and the location of each sensor can be described by four parameters. A generic shape-sensing modeling method is developed based on the 2-DOF bending deflections of the backbone, i.e., the relationship between the stretched lengths of two sensors and two joint variables. Due to the location errors of sensors, the calibration is conducted to further increase the accuracy of the shape-sensing model and realize the accurate shape measurement of the CDCR. The proposed shape-sensing method is insensitive to the location position errors of sensors, which can develop an accurate shape-sensing model with fewer sensors compared with the shape-sensing method applying the parallel sensor location scheme. Therefore, it has the advantages of high accuracy, high resolution ratio, and low cost. Based on the calibrated shape-sensing model, the sliding-mode-based closed-loop control is implemented for the CDCR.

The rest of this paper is organized as follows. Section 2 presents the configuration design of the CDCR with a pattern-based flexible backbone. Section 3 addresses the kinematic analysis of the CDCJM and the CDCR, including the displacement analysis, the velocity analysis, and the inverse kinematic analysis. Section 4 presents the shape-sensing modeling method of the CDCJM based on two stretchable capacitive sensors and the calibration method to compensate for the location errors of sensors. Furthermore, the closed-loop control is conducted for the CDCR based on the calibrated shape-sensing model. In Section 5, experiments on the CDCJM and the CDCR are conducted to validate the effectiveness of the proposed shape-sensing model and the closed-loop control method. The conclusion of this paper is given in Section 6.

2. Configuration Design of the Cable-Driven Continuum Robot

As shown in Figure 1, a CDCR usually employs a modular design method, which is composed of a set of identical serially connected CDCJMs. Each CDCJM merely allows 2-DOF bending motions, which consist of a base platform, a moving platform, a flexible backbone, and driving cables referring to Figure 2. The moving platform of the i^{th} CDCJM is the base platform of the $(i+1)^{\text{th}}$ CDCJM. The driving cables are evenly mounted on the base and moving platforms, respectively. The flexible backbone is fixed at the centers of the base and moving platforms.

In this paper, a pattern-based flexible backbone is employed for the CDCJM, which possesses low bending stiffness but high tensile stiffness and high torsion stiffness to achieve the designated 2-DOF bending motions of the CDCJM. The rectangular patterns employed are inspired by elastic couplings. As shown in Figure 3, each patterned segment has two rectangular patterns. The structure parameters of the pattern-based flexible backbone include the inner diameter d_1, the thickness t, the width of the pattern a, the distance between two adjacent patterns l, and the central angle subtended by the pattern β. Since the existing stiffness modeling methods of the patterned-based flexible backbone cannot develop accurate analytical stiffness models when the flexible backbone produces large bending deflections, an FEA-based data-driven parameter stiffness modeling method is proposed. Such a stiffness modeling method uses a set of structure parameters within their dimension bounds and the simulation stiffness values computed by the FEA software to train the stiffness model. The Gaussian Process Regression (GPR) is employed due to its capability of solving nonlinear regression problems with fewer training data. Based on the trained stiffness models, the structure parameter optimization of the flexible backbone is conducted using the particle swarm optimization method to minimize the ratio of bending stiffness to tensile stiffness and the ratio of bending stiffness to torsion stiffness. The optimized structure parameters are given as follows: $d_1 = 9\,\text{mm}$, $t = 4\,\text{mm}$, $l_1 = 1.1\,\text{mm}$, $a = 0.8\,\text{mm}$, and $\beta = 171.5°$.

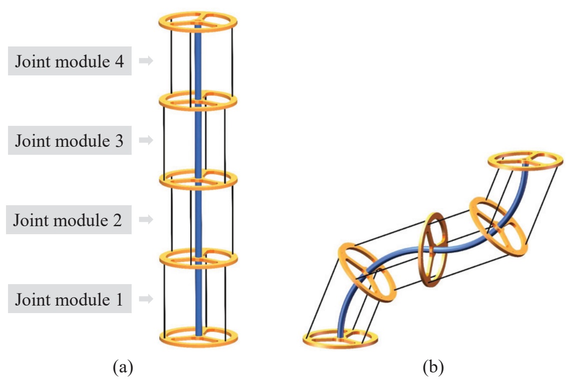


Figure 1. Schematic diagrams of a Cable-Driven Continuum Robot: (**a**) the initial pose without bending motions; (**b**) an arbitrary pose.

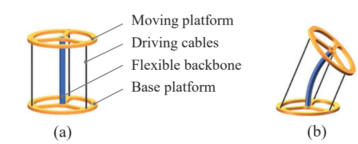


Figure 2. Schematic diagrams of a Cable-Driven Continuum Joint Module with three driving cables: (**a**) the initial pose without bending motions; (**b**) an arbitrary pose.

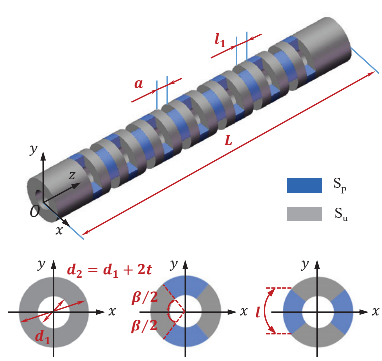


Figure 3. Schematic diagram and structure parameters of the pattern-based flexible backbone.

3. Kinematic Analysis of the Cable-Driven Continuum Robot

The kinematic analysis issues of the CDCR include the forward kinematics analysis, the differential kinematics analysis, and the inverse kinematics analysis. Since the CDCR can be considered as multiple CDCJMs connected in series, the kinematic analysis of the CDCR can be derived from the kinematic analysis of the CDCJM.

3.1. Kinematic Analysis of the Cable-Driven Continuum Joint Module

Two joint variables are introduced to simplify the kinematic analysis of the CDCJM, as shown in Figure 4. Based on the unique stiffness properties of low bending stiffness but high tensile and torsion stiffness, the bending shape of the optimized pattern-based flexible backbone can be considered as an arc in space with constant curvature. Therefore, the 2-DOF bending motions of the CDCJM can be expressed by two joint variables, including the bending direction angle $\alpha \in [0, 2\pi]$ and the bending angle $\theta \in [0, \pi/4]$.

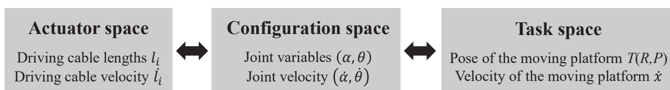


Figure 4. Three spaces and mapping of the Cable-Driven Continuum Joint Module.

Two coordinate systems, $\{B\}$ and $\{E\}$, are defined for the kinematic analysis of the CDCJM. The origins and axis directions of $\{B\}$ and $\{E\}$ are depicted in Figure 5. The attachment points of the i^{th} cable at the base and moving platforms are denoted by B_i and E_i, respectively. The plane O_bAO_e is the bending plane of the CDCJM. O_bB and O_eE are the intersecting lines of the bending plane with the base and moving platforms, respectively.

3.1.1. Displacement Analysis

The displacement analysis of the CDCJM is to derive the kinematic relationship between cable lengths and the pose of the moving platform. The relationship between driving cable lengths and the joint variables is computed by the closed-loop vector method. As shown in Figure 5, the i^{th} cable length is computed by the norm of the vector $\overrightarrow{B_iE_i}$:

$$\overrightarrow{B_iE_i} = \overrightarrow{B_iO_b} + \overrightarrow{O_bO_e} + \overrightarrow{O_eE_i} \tag{1}$$

where $\overrightarrow{B_iO_b}$, $\overrightarrow{O_bO_e}$, and $\overrightarrow{O_eE_i}$ are relative to the structure parameters and joint variables of the CDCJM. Their specific expressions can refer to [28].

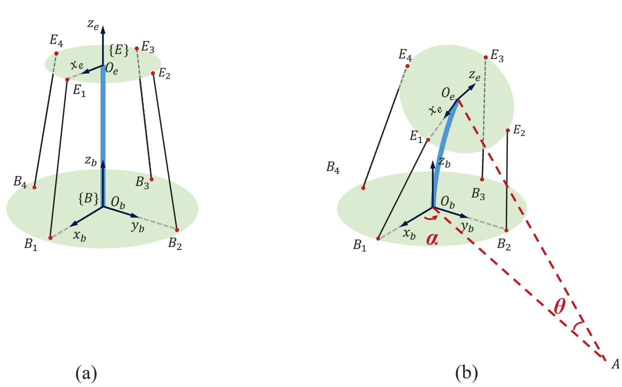


Figure 5. Kinematic diagrams of the Cable-Driven Continuum Joint Module: (**a**) the initial pose; (**b**) an arbitrary pose.

The analytical expression of the i^{th} cable length is given by

$$\|\overrightarrow{B_iE_i}\|^2 = (r_\mathrm{b} - r_\mathrm{m})^2 + 4\sin^2\frac{\theta}{2}\left(\frac{L}{\theta} - r_\mathrm{b}\cos\beta_i\right)\left(\frac{L}{\theta} - r_\mathrm{m}\cos\beta_i\right) \tag{2}$$

where L is the length of the flexible backbone. $\beta_i = \alpha + (i-1)\pi/2$ is the rotation angle from $\overrightarrow{O_bB_i}$ to $\overrightarrow{O_bB}$. r_b and r_m denote the attachment points of cables fixed on the base and moving platforms, respectively.

According to (2), the expressions of two joint variables are computed by

$$\alpha = \arctan\frac{l_2^2 - l_4^2}{l_3^2 - l_1^2} \tag{3}$$

$$\frac{(1-\cos\theta)}{\theta} = \frac{\sqrt{\left(l_3^2 - l_1^2\right)^2 + \left(l_2^2 - l_4^2\right)^2}}{4L(r_\mathrm{b} + r_\mathrm{m})} \tag{4}$$

Referring to [28], θ is given as

$$\theta = 3.1136 - \sqrt{9.6557 - 11.3636a} \tag{5}$$

where $a = \sqrt{\left(l_3^2 - l_1^2\right)^2 + \left(l_2^2 - l_4^2\right)^2}/(4L(r_\mathrm{b} + r_\mathrm{m}))$.

Based on the screw theory, the 2-DOF bending motions of the CDCJM can be described by the rotational movement around an instantaneous screw axis ξ. The rotational angle is θ. The pose of the moving platform is derived by the two-variable local Product-Of-Exponential (POE) formula

$$\boldsymbol{T}_{B,E}(\alpha,\theta) = e^{\hat{\boldsymbol{\xi}}\theta}\boldsymbol{T}_{B,E}(0) \tag{6}$$

where $\hat{\boldsymbol{\xi}} = \left[\begin{smallmatrix}\hat{\omega} & v \\ 0 & 0\end{smallmatrix}\right] \in se(3)$ is the twist of the CDCJM. ω and v represent the directional vector and the position vector of $\boldsymbol{\xi}$ with respect to frame B, respectively.

In (6), $\boldsymbol{T}_{B,E}(0)$ is the initial pose of CDCJM:

$$\boldsymbol{T}_{B,E}(0) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Referring to [28], v and ω are given by

$$\boldsymbol{v} = L\begin{bmatrix} -\frac{1}{2}\cos\alpha & -\frac{1}{2}\sin\alpha & \frac{1}{\theta} - \frac{1}{2}\cot\frac{1}{\theta} \end{bmatrix}^\mathrm{T} \tag{8}$$

$$\boldsymbol{\omega} = \begin{bmatrix} -\sin\alpha & \cos\alpha & 0 \end{bmatrix}^\mathrm{T} \tag{9}$$

According to (8) and (9), the screw axis is uniquely determined by the joint variables.

3.1.2. Velocity Analysis

As shown in Figure 4, the joint velocity is utilized for the velocity analysis of the CDCJM, in which the major issue is to compute the Jacobian matrix. The Jacobian matrix between the cable velocity and the joint velocity can be directly calculated according to (2), (3) and (5):

$$\boldsymbol{J}_{l\Phi} = \begin{bmatrix} \frac{\partial l_1}{\partial \alpha} & \frac{\partial l_2}{\partial \alpha} & \frac{\partial l_3}{\partial \alpha} & \frac{\partial l_4}{\partial \alpha} \\ \frac{\partial l_1}{\partial \theta} & \frac{\partial l_2}{\partial \theta} & \frac{\partial l_3}{\partial \theta} & \frac{\partial l_4}{\partial \theta} \end{bmatrix}^\mathrm{T} \tag{10}$$

Based on the Jacobian matrix $\boldsymbol{J}_{l\Phi}$, the cable velocity can be derived as

$$\mathrm{d}\boldsymbol{l} = \boldsymbol{J}_{l\Phi}\,\mathrm{d}\boldsymbol{\Phi} \tag{11}$$

where $\mathrm{d}\boldsymbol{l} = \begin{bmatrix} \mathrm{d}l_1 & \mathrm{d}l_2 & \mathrm{d}l_3 & \mathrm{d}l_4 \end{bmatrix}^{\mathrm{T}}$ is the cable velocity and $\mathrm{d}\boldsymbol{\Phi} = \begin{bmatrix} \mathrm{d}\alpha & \mathrm{d}\theta \end{bmatrix}^{\mathrm{T}}$ is the joint velocity.

Based on [28], the instantaneous spatial velocity of the moving platform can be given by

$$\hat{\boldsymbol{V}}_{B,E}^{B} = \dot{\boldsymbol{T}}_{B,E}(\alpha,\theta)\boldsymbol{T}_{B,E}^{-1}(\alpha,\theta) \tag{12}$$

where $\hat{\boldsymbol{V}}_{B,E}^{B} \in se(3)$ is a twist described in frame B, whose coordinates are formulated by $\boldsymbol{V}_{B,E}^{B} = (\boldsymbol{v}_{\mathrm{m}}^{B}, \boldsymbol{\omega}_{\mathrm{m}}^{B}) \in \Re^{6\times 1}$. $\boldsymbol{\omega}_{\mathrm{m}}^{B}$ and $\boldsymbol{v}_{\mathrm{m}}^{B}$ represent the instantaneous angular velocity and the linear velocity of the CDCJM, respectively. Their specific expressions are given in [28]. $\dot{\boldsymbol{T}}_{B,E}(\alpha,\theta)$ is the derivative of $\boldsymbol{T}_{B,E}(\alpha,\theta)$ with respect to the joint variables.

According to (12), the linear velocity of point O_e is derived by

$$\boldsymbol{v}_{O_e}^{B} = \boldsymbol{\omega}_{\mathrm{m}}^{B} \times \boldsymbol{p} + \boldsymbol{v}_{\mathrm{m}}^{B} \tag{13}$$

where $\boldsymbol{p} = \overrightarrow{O_b O_e}$.

Then, $\boldsymbol{J}_{x\Phi}$ is formulated as

$$\boldsymbol{J}_{x\Phi} = \begin{bmatrix} \dfrac{L\sin\alpha(\cos\theta - 1)}{\theta} & \dfrac{L\cos\alpha(\theta\sin\theta + \cos\theta - 1)}{\theta^2} \\ \dfrac{L\cos\alpha(1-\cos\theta)}{\theta} & \dfrac{L\sin\alpha(\theta\sin\theta + \cos\theta - 1)}{\theta^2} \\ 0 & \dfrac{L(-\sin\theta + \theta\cos\theta)}{\theta^2} \\ -\cos\alpha\sin\theta & -\sin\alpha \\ -\sin\alpha\sin\theta & \cos\alpha \\ 1-\cos\theta & 0 \end{bmatrix} \tag{14}$$

Therefore, the velocity of point O_e can be calculated as

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{J}_{x\Phi}\,\mathrm{d}\boldsymbol{\Phi} \tag{15}$$

where $\mathrm{d}\boldsymbol{x} = \left[\boldsymbol{v}_{O_e}^{B}, \boldsymbol{\omega}_{\mathrm{m}}^{B}\right]^{\mathrm{T}}$.

3.2. *Kinematic Analysis of the Cable-Driven Continuum Robot*

3.2.1. Forward Kinematic Analysis

Due to the modular design approach, the forward kinematic model of the CDCR is derived from the product of the forward kinematic models of the CDCJMs:

$$\boldsymbol{T}_{0,n}(\boldsymbol{\alpha},\boldsymbol{\theta}) = \boldsymbol{T}_{0,1}(\alpha_1,\theta_1)\cdots\boldsymbol{T}_{i-1,i}(\alpha_i,\theta_i)\cdots\boldsymbol{T}_{n-1,n}(\alpha_n,\theta_n) \tag{16}$$

According to (16), the pose of CDCR with respect to the base frame can be derived when given the number of the CDCJMs and their joint variables.

3.2.2. Differential Kinematic Analysis

Based on (16), the instantaneous spatial velocity of the CDCR is given by

$$\hat{\boldsymbol{V}}^{s} = \dot{\boldsymbol{T}}_{0,n}\boldsymbol{T}_{0,n}^{-1} = \dot{\boldsymbol{T}}_{0,1}\boldsymbol{T}_{0,1}^{-1} + \boldsymbol{T}_{0,1}\left(\dot{\boldsymbol{T}}_{1,2}\boldsymbol{T}_{1,2}^{-1}\right)\boldsymbol{T}_{0,1}^{-1} + \cdots + \boldsymbol{T}_{0,n-1}\left(\dot{\boldsymbol{T}}_{n-1,n}\boldsymbol{T}_{n-1,n}^{-1}\right)\boldsymbol{T}_{0,n-1}^{-1} \tag{17}$$

Substituting (12) into (17) and introducing the operator $\vee$, (17) can be rewritten as

$$\boldsymbol{V}^{s} = \boldsymbol{V}_{0,1}^{1} + Ad_{\boldsymbol{T}_{0,1}}\boldsymbol{V}_{1,2}^{2} + \cdots + Ad_{\boldsymbol{T}_{0,n-1}}\boldsymbol{V}_{n-1,n}^{n} \tag{18}$$

where the operator $\vee$ denotes the mapping from $se(3)$ to $\Re^{6\times 1}$. $\boldsymbol{V}^s = (\boldsymbol{v}^s, \boldsymbol{\omega}^s) \in \Re^{6\times 1}$ is the twist coordinate of $\hat{V}^s$, in which $\boldsymbol{\omega}^s$ and $\boldsymbol{v}^s$ represent the spatial angular velocity and the spatial linear velocity of the CDCR with respect to its base frame, respectively.

$Ad_{\boldsymbol{T}_{0,i}} \in \Re^{6\times 6}$ is the adjoint transformation of $\boldsymbol{T}_{0,i}$:

$$Ad_{\boldsymbol{T}_{0,i}} = \begin{bmatrix} \boldsymbol{R}_{0,i} & \boldsymbol{p}_{0,i}\boldsymbol{R}_{0,i} \\ \boldsymbol{0}_{3\times 3} & \boldsymbol{R}_{0,i} \end{bmatrix} \tag{19}$$

Therefore, (18) can be represented by

$$\boldsymbol{V}^s = \begin{bmatrix} \boldsymbol{J}_1 & Ad_{T_{0,1}}\boldsymbol{J}_2 & \cdots & Ad_{T_{0,n-1}}\boldsymbol{J}_n \end{bmatrix}\dot{\phi} = \boldsymbol{J}^s\dot{\phi} \tag{20}$$

where $\boldsymbol{J}_n$ is the spatial Jacobian matrix of the i^{th} CDCJM. $\dot{\boldsymbol{\phi}} = \left[\dot{\boldsymbol{\phi}}_1^{\text{T}}, \cdots, \dot{\boldsymbol{\phi}}_n^{\text{T}}\right]^{\text{T}}$ is the joint velocity of the CDCR.

3.2.3. Inverse Kinematic Analysis

The inverse kinematic analysis of the CDCR is to calculate the driving cable lengths when given the tip pose of the CDCR, which is significant for the trajectory planning of the CDCR. Referring to Figure 4, the inverse kinematic analysis of the CDCR is divided into two steps. The driving cable lengths can be calculated through (2) when given the joint variables of the CDCR. The analytical expression of the joint variables relative to the tip pose of the CDCR is difficult to derive since the CDCR is a hyper-redundant robot.

In this paper, the Newton-Rapson iteration method is employed. Given the desired pose $\boldsymbol{T}_{0,n}^{\text{d}}$ of the CDCR and the initial guess of joint variables $\boldsymbol{\phi}_0 = \left[\boldsymbol{\alpha}^0, \boldsymbol{\theta}^0\right]^{\text{T}}$, the pose $\boldsymbol{T}_{0,n}^0$ of the CDCR based on the initial guess of joint variables is derived through (16). The derivation between $\boldsymbol{T}_{0,n}^0$ and $\boldsymbol{T}_{0,n}^{\text{d}}$ is given by

$$\mathrm{d}\boldsymbol{x} = \left(\log T_{0,n}^{\text{d}}\left(T_{0,n}^0\right)^{-1}\right)^{\vee} \tag{21}$$

Based on the differential kinematics analysis, the differential changes in joint variables are computed by

$$\mathrm{d}\boldsymbol{\phi} = (\boldsymbol{J}^s)^{+}\mathrm{d}\boldsymbol{x} \tag{22}$$

Then, the joint variables are updated as

$$\boldsymbol{\phi}^{i+1} = \boldsymbol{\phi}^i + \mathrm{d}\boldsymbol{\phi}^{i+1} \tag{23}$$

where the right superscripts represent the iterations.

According to (21)–(23), the joint variables are constantly updated until the error between the desired pose and the tip pose computed by the updated joint variables is within the allowable range.

4. Shape Sensing of the Cable-Driven Continuum Robot

4.1. Stretchable Capacitive Sensor

To achieve accurate motion control for the CDCR, a closed-loop motion control scheme is employed for each CDCJM, in which sensor feedback for the bending deflection of the flexible backbone, i.e., the bending angle and the bending directional angle, is essential. Since the flexible backbone of the CDCJM can achieve a large bending angle of 45° with unlimited bending direction angles, the sensor has to possess the properties of a high stretch rate and high sensitivity in order to accurately detect the two joint angles. Among various strain sensors, the stretchable capacitive sensors made of flexible fabric and conductive nanomaterial are a class of elastic strain sensors with high sensitivity and high linearity for strain detection. As such, the stretchable capacitive sensor of Model RH-ESSA-01 from ElasTech is employed in this work, which has a maximum stretch rate of 50% and a minimum resolution of 0.05%.

The deflection of the stretchable capacitive sensor is the stretched deflection along its longitude direction. With the stretched deflection, the capacitance of the sensor will change. The employed stretchable sensors shown in Figure 6 have a length of 50 mm and a width of 20 mm, in which the effective sensing length is 40 mm. The relationship between the stretched length of the sensor and its corresponding capacitance is given in Figure 7. The result verifies the employed stretchable sensor has high linearity. The analytical expression through the curve fitting is given by

$$C = 2.082\Delta l_s + 61.75 \tag{24}$$

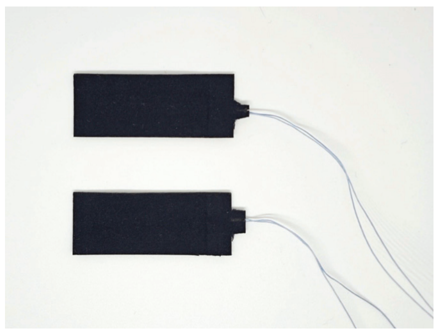


Figure 6. Prototypes of the stretchable capacitive sensors.

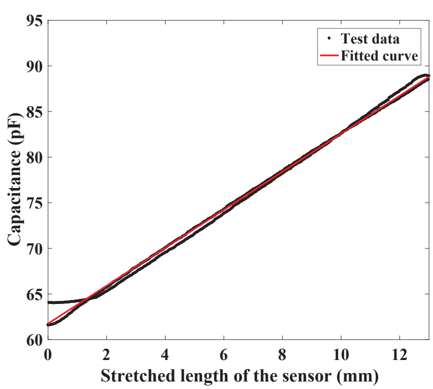


Figure 7. Relationship between the stretched length of a stretchable capacitive sensor and its corresponding capacitance.

4.2. Shape-Sensing Model of the Cable-Driven Continuum Joint Module

The shape-sensing model of the CDCJM is to derive the analytical expressions between the stretched lengths of stretchable sensors and the joint variables α and θ of the CDCJM. In this paper, a generic shape-sensing modeling method is proposed for the CDCJM, in which two joint variables can be measured through the stretched lengths of two stretchable capacitive sensors. Two stretchable capacitive sensors are located in arbitrary positions on the surface of the flexible backbone, and the location of each sensor is defined by its two endpoints. The width of the sensors is ignored in this paper, and each sensor can be simplified as a straight line. Since the surface of the flexible backbone without bending is a cylinder, the sensor can be considered as a helix along the cylinder, as shown in Figure 8. When the CDCJM produces the bending motions, the shape of the flexible backbone becomes a part of the torus, as shown in Figure 9. The location of each sensor varies from the bending deflections of the flexible backbone. The stretched length of a sensor is used to calculate the variation in the distance between two endpoints of the sensor on the surface of the backbone.

As shown in Figure 8, the two endpoints of the sensor are denoted by S_1 and S_2, respectively. An arbitrary point on the sensor is denoted by point S. Frame $\{B\}$ referring to Figure 5 is employed as the base frame of the flexible backbone. The intersection point between the x_b axis of frame $\{B\}$ and the backbone surface is denoted by point A. The flexible backbone is unfolded into a plane $ABB'A'$ along the straight line AB. A rectangular coordinate system is established at point A. The helix in the plane $ABB'A'$ is a straight line S_1S_2, in which the angle between the x-axis and the line S_1S_2 is denoted by λ and the intercept of the line S_1S_2 on the y-axis is denoted by b. The helix is projected onto the bottom of the flexible backbone and the projected arc is denoted by $\overset{\frown}{S_1'S_2'}$. The projection point of point S is denoted by S'. $t \in [t_0, t_0 + \Delta t]$ is the angle between the vector $\overrightarrow{O_bS'}$ and the x_b-axis, in which t_0 is the angle between the vector $\overrightarrow{O_bS_1'}$ and the x_b-axis. Therefore, the location of the sensor is defined by four parameters, i.e., b, λ, t_0, and Δt. b and t_0 are employed to determine the endpoint position of the sensor. λ is employed to describe the attachment angle of the sensor. Δt is employed to represent the length of the sensor. $\boldsymbol{a} = [b, \lambda, t_0, \Delta t]$ is produced to denote the location of the sensor.

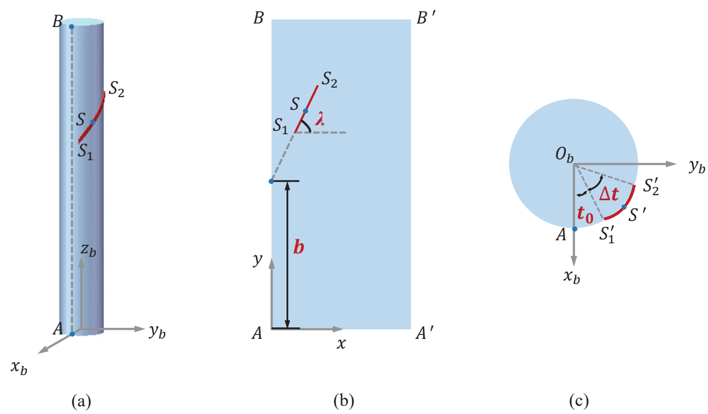


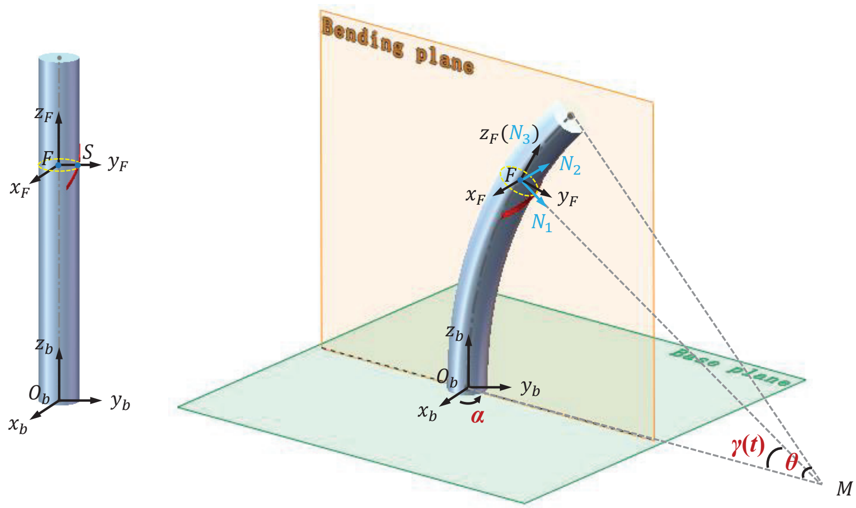
Figure 8. Schematic diagrams of the flexible backbone without bending and the corresponding projection views: (**a**) model of the flexible backbone attached with a stretchable capacitive sensor; (**b**) two-dimensional unfolded surface of the flexible backbone; (**c**) bottom plane of the flexible backbone and the corresponding projection of the sensor.

As shown in Figure 9, the intersection between the plane parallel to the bottom of the flexible backbone at point S and the neutral axis of the flexible backbone is denoted by point F. A local coordinate system $\{F\}$ is established at point F, in which the axes of frame $\{F\}$ are consistent with those of frame $\{B\}$ when the flexible backbone is in the initial state. The coordinates of the point S with respect to frame $\{F\}$ and frame $\{B\}$ are given by

$$\boldsymbol{P}_S^F(0) = [r\cos(t), r\sin(t), 0]^{\mathrm{T}} \tag{25}$$

$$\boldsymbol{P}_S^B(0) = \begin{bmatrix} x_S^B(0) \\ y_S^B(0) \\ z_S^B(0) \end{bmatrix} = \begin{bmatrix} r\cos t \\ r\sin t \\ r\cdot t\tan\lambda + b \end{bmatrix} \tag{26}$$

where $r = (d_1 + 2t)/2$ is the radius of the flexible backbone.



(a) (b)

Figure 9. Schematic diagrams of the flexible backbone with a stretchable capacitive sensor in its bending state: (**a**) model of the flexible backbone without bending; (**b**) model of the flexible backbone with the bending deflection.

When the flexible backbone produces the bending deflection, a coordinate system $\{N\}$ is introduced as a local frame, in which the origin is located at point F. The $\boldsymbol{N}_1$ axis points to the center of curvature M and the $\boldsymbol{N}_3$ axis is tangent to the neutral axis of the flexible backbone. The coordinates of point S expressed in frame $\{N\}$ becomes

$$\boldsymbol{P}_S^N(\alpha) = [r\cos(t-\alpha), r\sin(t-\alpha), 0]^{\mathrm{T}} \tag{27}$$

The unit directional vectors of the $\boldsymbol{N}_1$, $\boldsymbol{N}_2$, and $\boldsymbol{N}_3$ axes described in frame $\{B\}$ are derived by

$$\boldsymbol{N}_1 = \begin{bmatrix} \cos\alpha\cos(\gamma(t)) \\ \sin\alpha\cos(\gamma(t)) \\ -\sin(\gamma(t)) \end{bmatrix} \tag{28}$$

$$\boldsymbol{N}_2 = \begin{bmatrix} -\sin\alpha \\ \cos\alpha \\ 0 \end{bmatrix} \tag{29}$$

$$\boldsymbol{N}_3 = \boldsymbol{N}_1 \times \boldsymbol{N}_2 \tag{30}$$

Therefore, the orientation matrix of frame $\{N\}$ is given by

$$N = [\boldsymbol{N}_1, \boldsymbol{N}_2, \boldsymbol{N}_3] \tag{31}$$

Based on the constant curvature assumption, the coordinate of point F described in frame $\{B\}$ is calculated by

$$\boldsymbol{P}_F^B(\alpha,\theta) = \begin{bmatrix} x_F^B \\ y_F^B \\ z_F^B \end{bmatrix} = \begin{bmatrix} R(1-\cos(\gamma(t)))\cos\alpha \\ R(1-\cos(\gamma(t)))\sin\alpha \\ R\sin(\gamma(t)) \end{bmatrix} \tag{32}$$

where $R = L/\theta$ is the radius of the backbone bending curvature. $\gamma(t) = z_0\theta/L = (r \cdot t\tan\lambda + b)\theta/L$ is the angle between the line FM and the line O_bM.

According to (27), (32), and (31), the coordinates of the point S with respect to frame $\{B\}$ are derived by

$$\begin{aligned} \boldsymbol{P}_S^B(\alpha,\theta) &= \boldsymbol{N}\cdot\boldsymbol{P}_S^N(\alpha)+\boldsymbol{P}_F^B(\alpha,\theta) \\ &= \begin{bmatrix} r\cos(-t+\alpha)\cos(\alpha)\cos(\gamma(t))+r\sin(-t+\alpha)\sin(\alpha)+ \\ R(1-\cos(\gamma(t)))\cos(\alpha) \\ r\cos(-t+\alpha)\sin(\alpha)\cos(\gamma(t))-r\sin(-t+\alpha)\cos(\alpha)+ \\ R(1-\cos(\gamma(t)))\sin(\alpha) \\ -r\cos(-t+\alpha)\sin(\gamma(t))+R\sin(\gamma(t)) \end{bmatrix} \end{aligned} \tag{33}$$

Based on the geometrical relationship shown in Figure 8, the initial length of the sensor is derived by

$$l_s(0)=\frac{\Delta\mathrm{t}\cdot r}{\cos\lambda} \tag{34}$$

where $l_s(0)$ represents the length of the sensor in the initial state.

Differentiating the both sides of (33), it becomes

$$\begin{aligned} \left(\frac{d\boldsymbol{P}_S^B}{\mathrm{d}t}\right)^2 &= \left(\frac{\partial x}{\partial t}\right)^2+\left(\frac{\partial y}{\partial t}\right)^2+\left(\frac{\partial z}{\partial t}\right)^2 \\ &= \frac{(\tan\lambda)^2\theta^2r^2\left(\frac{L}{\theta}-r\cos(t-\alpha)\right)^2}{L^2}+r^2 \\ &= (\tan\lambda)^2r^2\left(1-\frac{r}{L}\theta\cos(t-\alpha)\right)^2+r^2 \\ &= r^2\cdot\left((\tan\lambda)^2(1-k\theta\cos(t-\alpha))^2+1\right) \end{aligned} \tag{35}$$

where $k=r/L$.

Therefore, the stretched length of the sensor relative to α and θ is calculated by

$$\begin{aligned} \Delta l_{\mathrm{s}} &= \int_{t_0}^{t_0+\Delta\mathrm{t}} d\boldsymbol{P}_S^B\mathrm{d}t-l_s(0) \\ &= r\int_{t_0}^{t_0+\Delta\mathrm{t}}\left(\sqrt{(\tan\lambda)^2(1-k\theta\cos(t-\alpha))^2+1}-\frac{1}{\cos\lambda}\right)\mathrm{d}t \end{aligned} \tag{36}$$

According to (36), the numerical iteration method is employed to solve the joint variables when given the stretched lengths of sensors. Therefore, it needs to solve the Jacobian matrix between the stretched lengths of sensors and the joint variables. Differentiating both sides of (36), it becomes

$$\frac{\partial\Delta l_{\mathrm{s}}}{\partial\alpha}=r(g(\theta,\alpha,\lambda,t_0,k)-g(\theta,\alpha,\lambda,t_0+\Delta\mathrm{t},k)) \tag{37}$$

$$\frac{\partial\Delta l_{\mathrm{s}}}{\partial\theta}=r\left(\int_{t_0}^{t_0+\Delta\mathrm{t}}-\frac{\left(g(\theta,\alpha,\lambda,t,k)^2-1\right)k\cos(t-\alpha)}{g(\theta,\alpha,\lambda,t,k)}\mathrm{d}t\right) \tag{38}$$

where $\mathrm{g}(\theta,\alpha,\lambda,t,k)=\sqrt{(\tan\lambda)^2(1-k\theta\cos(t-\alpha))^2+1}$.

Assume that the number of stretchable capacitive sensors is m. The Jacobian matrix between the stretched lengths of sensors and the joint variables is defined as

$$\boldsymbol{J}_a = \begin{bmatrix} \dfrac{\partial \Delta l_{s1}}{\partial \alpha} & \dfrac{\partial \Delta l_{s1}}{\partial \theta} \\ \dfrac{\partial \Delta l_{s2}}{\partial \alpha} & \dfrac{\partial \Delta l_{s2}}{\partial \theta} \\ \vdots & \vdots \\ \dfrac{\partial \Delta l_{sm}}{\partial \alpha} & \dfrac{\partial \Delta l_{sm}}{\partial \theta} \end{bmatrix} \tag{39}$$

where $\Delta l_{si}(i = 1, 2, ..., m)$ represents the tensile length of the i^{th} sensor.

When the flexible backbone produces the bending deflection, the stretched lengths of sensors $\boldsymbol{l}_s^a = \left[l_{s1}^a, \cdots, l_{sm}^a\right]^{\text{T}}$ are derived according to their capacitances. Given the initial guess of joint variables, the initial stretched lengths of sensors $\boldsymbol{l}_s^0 = \left[l_{s1}^0, \cdots, l_{sm}^0\right]^{\text{T}}$ are calculated based on (36). Then, the derivation between $\boldsymbol{l}_s^0$ and $\boldsymbol{l}_s^a$ is calculated. The differential changes in joint variables are given by

$$\mathrm{d}\boldsymbol{\phi} = (\boldsymbol{J}_a)^{+}\mathrm{d}\boldsymbol{l}_s \tag{40}$$

The joint variables are updated referring to (23) until $\mathrm{d}\Delta\boldsymbol{l}_s$ is within the allowable range. In order to further verify the feasibility of the proposed shape-sensing model based on the numerical iteration method, a computation example is provided. In this example, two stretchable capacitive sensors are employed, such that $\boldsymbol{J}_a \in \Re^{2\times 2}$. The structure parameters of the flexible backbone and the locations of two sensors are given in Table 1.

Table 1. Structure parameters of the flexible backbone and the locations of the sensors for the computation example.

Property	Value
Flexible backbone length L	120 mm
Flexible backbone radius r	10 mm
Attachment position of the first sensor $\boldsymbol{a}_1$	[20 mm, $5\pi/12$ rad, $\pi/2$ rad, $\pi/3$ rad]
Attachment position of the second sensor $\boldsymbol{a}_2$	[50 mm, $2\pi/5$ rad, $\pi/12$ rad, $\pi/3$ rad]

The stretched lengths of the two sensors are given by $\Delta l_{s1} = -3.15$ mm and $\Delta l_{s2} = -3.39$ mm. The initial guesses of the joint variables are $\alpha = 0.76$ rad and $\theta = 1.07$ rad. The average stretched length error of two sensors is calculated and its convergence result is shown in Figure 10. The updated joint variables are $\alpha = 1.26$ rad and $\theta = 1.57$ rad.

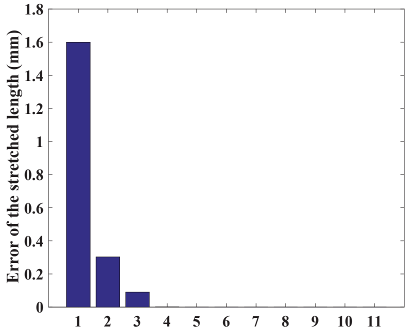


Figure 10. Convergence result of the stretched length error.

4.3. Calibration of Locations for Stretchable Capacitive Sensors

Due to the location errors of sensors, there exist derivations between the nominal stretched lengths and the actual stretched lengths, which will decrease the accuracy of the developed shape-sensing model. Therefore, it is necessary to calibrate the locations of sensors. Referring to (36), the stretched lengths of the sensors are determined by λ, t_0, and Δt. $\boldsymbol{d} = [\lambda, t_0, \Delta t]$ is produced to denote the stretched parameters of the sensor that need to be calibrated. Based on (36), the derivatives of the stretched length for a sensor relative to λ, t_0, and Δt are calculated by

$$\frac{\partial \Delta l_s}{\partial \lambda} = r\left(\int_{t_0}^{t_0+\Delta t}\left(\frac{\tan\lambda(1-k\theta\cos(-t+\alpha))^2\left((\tan\lambda)^2+1\right)}{g(\theta,\alpha,\lambda,t,k)} - \frac{\sin\lambda}{(\cos\lambda)^2}\right)\mathrm{d}t\right) \tag{41}$$

$$\frac{\partial \Delta l_s}{\partial t_0} = r(g(\theta,\alpha,\lambda,t_0+\Delta t,k) - g(\theta,\alpha,\lambda,t_0,k)) \tag{42}$$

$$\frac{\partial \Delta l_s}{\partial \Delta_t} = r(g(\theta,\alpha,\lambda,t_0+\Delta t,k) - \frac{1}{\cos\lambda}) \tag{43}$$

The Jacobian matrix of the sensor stretched lengths relative to λ, t_0, and Δt is given by

$$\boldsymbol{J}_b = \begin{bmatrix} \frac{\partial \Delta l_{s1}}{\partial \lambda} & \frac{\partial \Delta l_{s1}}{\partial t_0} & \frac{\partial \Delta l_{s1}}{\partial \Delta t} \\ \frac{\partial \Delta l_{s2}}{\partial \lambda} & \frac{\partial \Delta l_{s2}}{\partial t_0} & \frac{\partial \Delta l_{s2}}{\partial \Delta t} \\ \vdots & \vdots & \vdots \\ \frac{\partial \Delta l_{sm}}{\partial \lambda} & \frac{\partial \Delta l_{sm}}{\partial t_0} & \frac{\partial \Delta l_{sm}}{\partial \Delta t} \end{bmatrix} \tag{44}$$

Similarly, the numerical iteration method is utilized to derive the actual locations of sensors based on (44).

4.4. Closed-Loop Control of the Cable-Driven Continuum Robot

Based on the calibrated shape-sensing model, closed-loop control is conducted for the CDCR, as shown in Figure 11. In this paper, sliding mode control is employed as the control method for the CDCR, which is a robust control method that can effectively solve the control problem under parameter uncertainty.

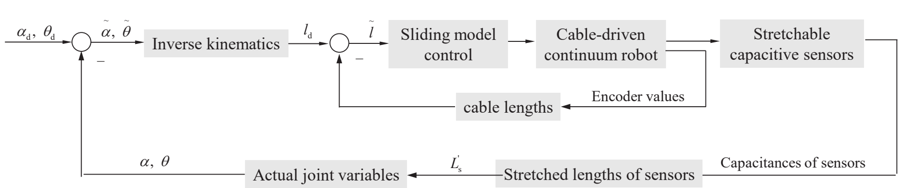


Figure 11. Block diagram of the closed-loop control method for the Cable-Driven Continuum Robot.

For the CDCR, the dynamic equation expressed in the motor frame is given by

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{f}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{u} \tag{45}$$

where $\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}$, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$, and $\boldsymbol{G}(\boldsymbol{q})$ are the generalized mass term, coriolis and centrifugal force terms, and the gravitational force term of the CDCR, respectively. $\boldsymbol{f}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is the frictional force term of the CDCR. $\boldsymbol{u}$ is the actuation force term of the motors. $\boldsymbol{q}$ is the rotation angle vector of the motors.

Given the desired pose $\boldsymbol{x}_\mathrm{d}$ of the CDCR, the desired rotation angle vector $\boldsymbol{q}_\mathrm{d}$ can be computed based on the inverse kinematic analysis. The rotation angle errors of the motors are denoted by $\widetilde{\boldsymbol{q}} = \boldsymbol{q}_\mathrm{d} - \boldsymbol{q}$. Therefore, (45) can be rewritten as

$$\begin{aligned} \boldsymbol{M}(\boldsymbol{q})\ddot{\widetilde{\boldsymbol{q}}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\widetilde{\boldsymbol{q}}} &= \boldsymbol{G}(\boldsymbol{q}) - \boldsymbol{f}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \boldsymbol{u} + \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_\mathrm{d} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}_\mathrm{d} \\ &= \boldsymbol{\eta} - \boldsymbol{u} \end{aligned} \tag{46}$$

The sliding mode surface is given as

$$\boldsymbol{s} = \boldsymbol{c}\widetilde{\boldsymbol{q}} + \dot{\widetilde{\boldsymbol{q}}} \tag{47}$$

$$\dot{\boldsymbol{s}} = \boldsymbol{c}\dot{\widetilde{\boldsymbol{q}}} + \ddot{\widetilde{\boldsymbol{q}}} \tag{48}$$

where $\boldsymbol{c} \in \Re^{m\times m}$ is a diagonal matrix. m is the number of motors.

Substituting (48) into (46), it becomes

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{s}} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{c}\dot{\widetilde{\boldsymbol{q}}} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\widetilde{\boldsymbol{q}}} - \boldsymbol{u} + \boldsymbol{\eta} \tag{49}$$

Let $\boldsymbol{\mu} = \boldsymbol{\eta} - \boldsymbol{M}(\boldsymbol{q})\boldsymbol{c}\dot{\widetilde{\boldsymbol{q}}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{c}\widetilde{\boldsymbol{q}} + \boldsymbol{s}$, (49) becomes

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{s}} = -\boldsymbol{s} - \boldsymbol{u} + \boldsymbol{\mu} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{s} \tag{50}$$

The Lyapunov function is constructed as

$$\boldsymbol{V} = \frac{1}{2}\boldsymbol{s}^\mathrm{T}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{s} \tag{51}$$

According to (51), its differential equation is calculated as

$$\dot{\boldsymbol{V}} = \boldsymbol{s}^\mathrm{T}\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{s}} + \frac{1}{2}\boldsymbol{s}^\mathrm{T}\dot{\boldsymbol{M}}\boldsymbol{s} \tag{52}$$

Let $\boldsymbol{u} = k\mathrm{sup}(\boldsymbol{s}) + \boldsymbol{k_p s}$, $\dot{\boldsymbol{V}}$ is positive when $k \geq \sup\|\boldsymbol{\mu}\|$.

5. Experimental Results

5.1. Experimental Verification of the Cable-Driven Continuum Joint Module

The experiment of the CDCJM includes the stability measurement of the stretchable sensors and the accuracy test of the developed shape-sensing model under the proposed calibration method. The experimental prototype of the CDCJM is shown in Figure 12, which consists of motors, the prototype of the CDCJM, the motion capture system, and two stretchable sensors. The Qualisys motion capture system includes six cameras and markers, which have a measurement accuracy of 0.32 mm. The type of camera is an Opus 500. The markers are evenly fixed on the base platform and the moving platform of the CDCJM, in which the coordinates of the markers are measured to calculate the pose of the CDCJM. The transformation matrix $\boldsymbol{T}_E^B$ is calculated by

$$\boldsymbol{T}_E^B = \boldsymbol{T}_E^\mathrm{cam}(\boldsymbol{T}_B^\mathrm{cam})^{-1} \tag{53}$$

where $\boldsymbol{T}_E^\mathrm{cam}$ and $\boldsymbol{T}_B^\mathrm{cam}$ represent the transformation matrices of frame $\{E\}$ frame $\{B\}$ with respect to the camera coordinate system, respectively.

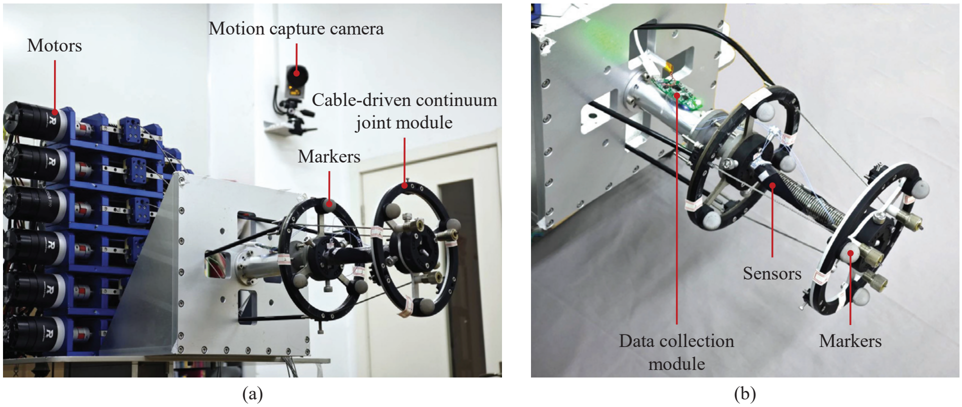


Figure 12. Experimental setup of the Cable-Driven Continuum Joint Module: (**a**) pose measurement system with motion capture cameras; (**b**) prototype of the Cable-Driven Continuum Joint Module with two stretchable capacitive sensors.

In order to measure the stability of the stretchable capacitive sensors, the moving platform of the CDCJM follows a circle trajectory, in which the bending angle θ is a constant of 0.3 rad and the bending direction angle α varies from 0 rad to 2π rad. The nominal locations of two stretchable capacitive sensors are $\boldsymbol{d}_{\text{nom1}} = [4\pi/9, \pi/2, \pi/2]$ rad and $\boldsymbol{d}_{\text{nom2}} = [-\pi/3, 3\pi/4, \pi]$ rad, respectively. The initial length of each sensor is 50 mm, and its prestretch length is 6 mm. The capacitance changes in two stretchable sensors are shown in Figure 13. With the bending motions of the CDCJM, the capacitances of the two stretchable sensors vary with their stretched lengths. When the CDCJM returns to its initial pose, the capacitance changes in the two stretchable sensors are zero.

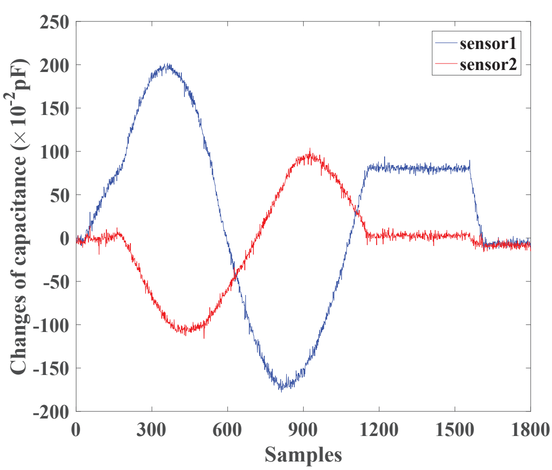


Figure 13. Capacitance of two stretchable sensors during the sensor stability test experiment.

Furthermore, the calibration of the locations for two stretchable sensors is conducted to increase the accuracy of the developed shape-sensing model. When given the nominal joint variables, the nominal stretched lengths of two sensors and their corresponding capacitance changes can be calculated according to (36) and (24), respectively. Three experimental poses are shown in Figure 14. The actual capacitance changes are measured through the data collection module. The capacitance errors of two sensors between the nominal values and the measured values are shown in Figures 15 and 16, which results from the location errors of the sensors. The average capacitance errors of the two sensors before calibration are 0.4433 pF and 0.3916 pF, respectively. Based on the numerical iteration method, the calibrated locations of two stretchable sensors are $\boldsymbol{d}_{c1} = [1.3897, 1.1887, 1.5708]$ rad and $\boldsymbol{d}_{c2} = [-1.1545, 2.9663, 3.1416]$ rad, respectively. According to the calibrated parameters, the capacitance errors of the two sensors after calibration are reduced to 0.1123 pF and 0.0679 pF, respectively. Based on the calibrated shape-sensing model, the verification trajectory is introduced, in which the bending angle θ is a constant of 0.15 rad and the bending direction angle α varies from 0 rad to 2π rad. The capacitance errors of the two sensors between the calibrated values and the actual values after calibration are 0.0988 pF and 0.0367 pF, respectively. The errors of α and θ between the measured values and the actual values are about 0.03 rad and 0.01 rad, respectively.

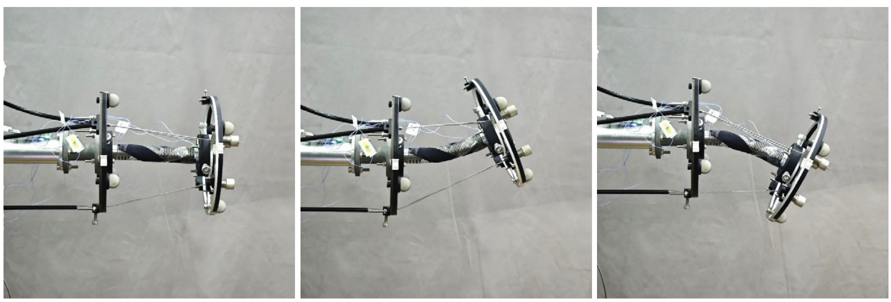


Figure 14. Experimental poses of the Cable-Driven Continuum Joint Module.

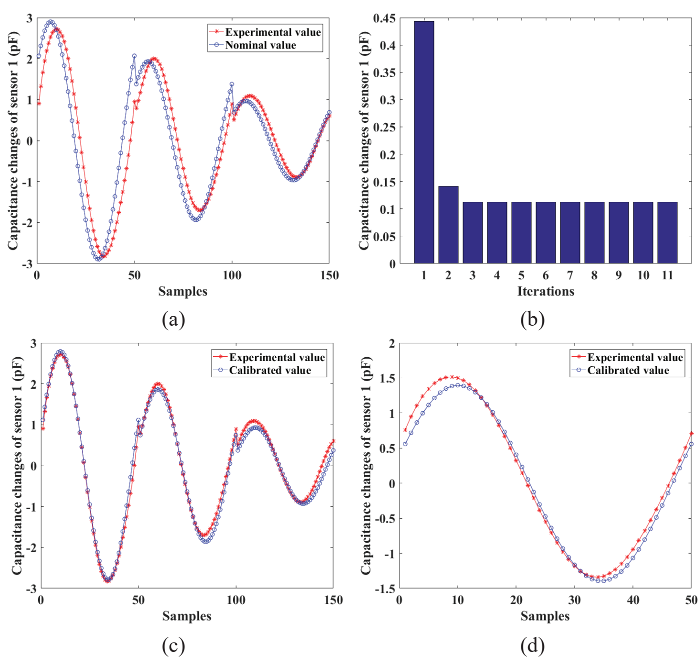


Figure 15. Calibration results of the first stretchable sensor: (**a**) capacitance errors before calibration; (**b**) convergence result of the capacitance errors; (**c**) capacitance errors after calibration; (**d**) capacitance errors of the verification trajectory after calibration.

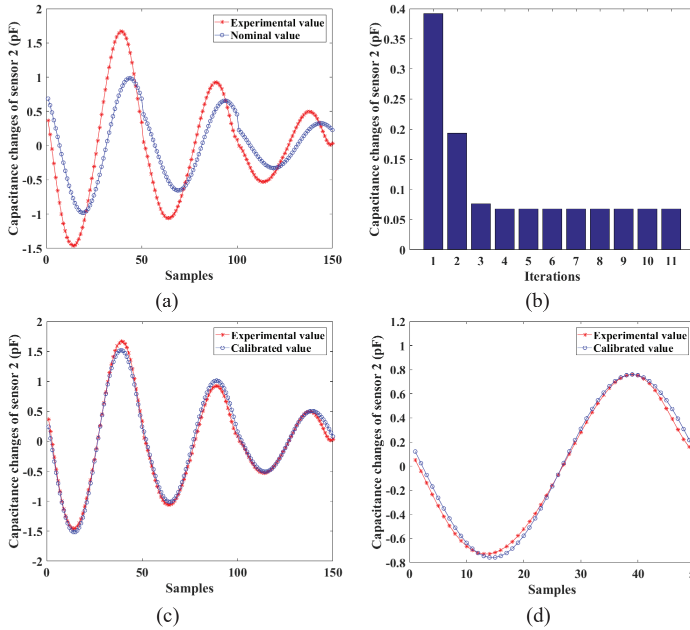


Figure 16. Calibration results of the second stretchable sensor: (**a**) capacitance errors before calibration; (**b**) convergence result of the capacitance errors; (**c**) capacitance errors after calibration; (**d**) capacitance errors of the verification trajectory after calibration.

5.2. Experimental Verification of the Cable-Driven Continuum Robot

In order to verify the effectiveness of the kinematics control method based on the calibrated shape-sensing model, the experiment of the CDCR is conducted. The prototype of the CDCR with multiple stretchable sensors is shown in Figure 17, which is composed of four CDCJMs. The flexible backbone of each CDCJM employs two stretchable sensors to measure its joint variables. The trajectory tracking accuracy of the CDCR is measured based on a circle trajectory, in which the radius of the circle is 55 mm. As shown in Figure 18, the tracking errors of the CDCR under the open-loop control and the closed-loop control are 49.23 and 8.40 mm, respectively. Compared with the tracking error under the open-loop control, the tracking error under the closed-loop control is reduced by 82.94%.

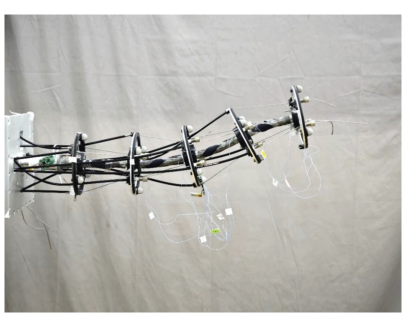


Figure 17. Experimental pose of the Cable-Driven Continuum Robot.

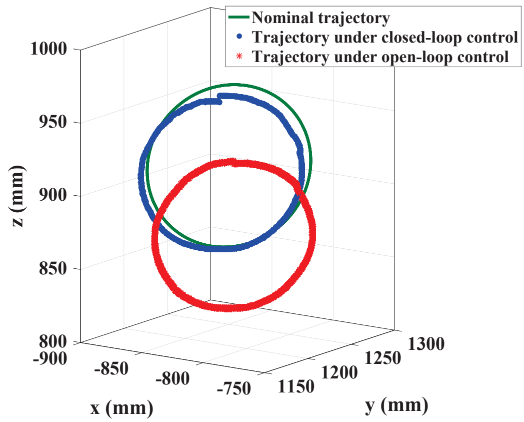


Figure 18. Trajectory tracking accuracy of the Cable-Driven Continuum Robot under open-loop control and closed-loop control.

6. Conclusions

This paper proposes a generic shape-sensing modeling method for a Cable-Driven Continuum Robot (CDCR) with a flexible backbone. For the Cable-Driven Continuum Joint Module (CDCJM) with 2-DOF bending motions, its joint variables can be measured through stretched lengths of two stretchable sensors. Combined with the calibration of the locations for the sensors, the accurate measurement of the joint variables can be realized. The proposed shape-sensing modeling method has the advantages of good stability, high resolution ratio, high accuracy, and good antidisturbance ability. Based on the calibrated shape-sensing model, the sliding-mode-based closed-control method is implemented for the CDCR. The accuracy of the calibrated shape-sensing model is measured during the experiments, in which the capacitance errors of the sensors between the calibrated values and the actual values are 0.0988 pF and 0.0367 pF, respectively. The corresponding errors of the two joint variables α and θ are about 0.03 rad and 0.01 rad, respectively. According to the experimental result, the tracking error of the CDCR is reduced from 49.23 mm to 8.40 mm under the closed-loop control, which verifies the effectiveness of the proposed kinematic control method based on the calibrated shape-sensing model.

Author Contributions: Conceptualization, W.S. and G.Y.; methodology, W.S., J.H. and X.K.; software, J.H.; validation, W.S. and J.H.; formal analysis, J.H. and X.K.; investigation, W.S.; resources, W.S. and H.B.; data curation, J.H.; writing—original draft preparation, W.S.; writing—review and editing, W.S.; visualization, W.S., J.H. and H.B.; supervision, G.Y.; project administration, G.Y. and Z.F.; funding acquisition, Z.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by The National Natural Science Foundation (Grant number: U1909215, 92048201, 52127803); the Basic Public Welfare Research Plan of Zhejiang Province (Grant number: LD22E050007, 2023C01176); the Ningbo Key Project of Scientific and Technological Innovation 2025 (Grant number: 2021Z020, 2021Z128).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CDCR	Cable-Driven Continuum Robot
CDCJM	Cable-Driven Continuum Joint Module
EM	ElectroMagnetic
FBG	Fiber Bragg Grating
SOM	Self Organizing Mapping
DOF	Degree Of Freedom
FEA	Finite Element Analysis
GPR	Gaussian Process Regression
POE	Product Of Exponential

References

1. Iii, R.J.W.; Jones, B.A. Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review. *Int. J. Robot. Res.* **2010**, *29*, 1661–1683.
2. Walker, I.D. Continuous backbone "continuum" robot manipulators. *Int. Sch. Res. Not.* **2013**, *2013*, 726506. [CrossRef]
3. Mishra, A.K.; Mondini, A.; Del Dottore, E.; Sadeghi, A.; Tramacere, F.; Mazzolai, B. Modular continuum manipulator: Analysis and characterization of its basic module. *Biomimetics* **2018**, *3*, 3. [CrossRef] [PubMed]
4. Burgner-Kahrs, J.; Rucker, D.C.; Choset, H. Continuum Robots for Medical Applications: A Survey. *IEEE Trans. Robot.* **2015**, *31*, 1261–1280. [CrossRef]
5. Lai, J.; Huang, K.; Lu, B.; Zhao, Q.; Chu, H.K. Verticalized-tip trajectory tracking of a 3d-printable soft continuum robot: Enabling surgical blood suction automation. *IEEE/ASME Trans. Mechatron.* **2021**, *27*, 1545–1556. [CrossRef]
6. Xu, K.; Zhao, J.; Fu, M. Development of the SJTU unfoldable robotic system (SURS) for single port laparoscopy. *IEEE/ASME Trans. Mechatron.* **2014**, *20*, 2133–2145. [CrossRef]
7. York, P.A.; Swaney, P.J.; Gilbert, H.B.; Webster, R.J. A wrist for needle-sized surgical robots. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 1776–1781.
8. Fukuda, J.; Konyo, M.; Takeuchi, E.; Tadokoro, S. Remote vertical exploration by active scope camera into collapsed buildings. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 1882–1888.
9. Wooten, M.; Frazelle, C.; Walker, I.D.; Kapadia, A.; Lee, J.H. Exploration and inspection with vine-inspired continuum robots. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 5526–5533.
10. Mehling, J.; Diftler, M.; Chu, M.; Valvo, M. A minimally invasive tendril robot for in-space inspection. In Proceedings of the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 20–22 February 2006; pp. 690–695.
11. Dong, X.; Axinte, D.; Palmer, D.; Cobos, S.; Raffles, M.; Rabani, A.; Kell, J. Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines. *Robot.-Comput.-Integr. Manuf.* **2017**, *44*, 218–229. [CrossRef]
12. Wang, M.; Dong, X.; Ba, W.; Mohammad, A.; Axinte, D.; Norton, A. Design, modelling and validation of a novel extra slender continuum robot for in-situ inspection and repair in aeroengine. *Robot.-Comput.-Integr. Manuf.* **2021**, *67*, 102054. [CrossRef]
13. Nakamoto, M.; Nakada, K.; Sato, Y.; Konishi, K.; Hashizume, M.; Tamura, S. Intraoperative Magnetic Tracker Calibration Using a Magneto-Optic Hybrid Tracker for 3-D Ultrasound-Based Navigation in Laparoscopic Surgery. *IEEE Trans. Med. Imaging* **2008**, *27*, 255–270. [CrossRef] [PubMed]
14. Song, S.; Li, Z.; Yu, H.; Ren, H. Electromagnetic Positioning for Tip Tracking and Shape Sensing of Flexible Robots. *IEEE Sensors J.* **2015**, *15*, 4565–4575. [CrossRef]
15. Luoa, X.; Wanc, Y.; Hec, X.; Moria, K. Observation-driven adaptive differential evolution and its application to accurate and smooth bronchoscope three-dimensional motion tracking. *Med. Image Anal.* **2015**, *24*, 282–296. [CrossRef] [PubMed]
16. Xu, F.; Wang, H.; Chen, W.; Miao, Y. Visual Servoing of a Cable-Driven Soft Robot Manipulator with Shape Feature. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4281–4288. [CrossRef]
17. Norouzi-Ghazbi, S.; Janabi-Sharifi, F. A Switching Image-Based Visual Servoing Method for Cooperative Continuum Robots. *J. Intell. Robot. Syst.* **2021**, *103*, 42. [CrossRef]
18. Li, J.; Zhang, F.; Yang, Z.; Jiang, Z.; Wang, Z.; Liu, H. Shape Sensing for Continuum Robots by Capturing Passive Tendon Displacements with Image Sensors. *IEEE Robot. Autom. Lett.* **2022**, *7*, 3130–3137. [CrossRef]
19. Searle, T.C.; Althoefer, K.; Seneviratne, L.; Liu, H. An optical curvature sensor for flexible manipulators. In Proceedings of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013.
20. Malekzadeh, M.; Mustafa, G.; Kwon, I.-B.; Catbas, N. An integrated approach for structural health monitoring using an in-house built fiber optic system and non-parametric data analysis. *Smart Struct. Syst.* **2014**, *14*, 917–942. [CrossRef]
21. Chitalia, Y.; Deaton, N.; Jeong, S.; Rahman, N.; Desai, J. Towards FBG-based Shape Sensing for Micro-scale and Meso-Scale Continuum Robots with Large Deflection. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1712–1719. [CrossRef] [PubMed]

22. Huang, J.; Huo, W.; Xu, W.; Mohammed, S.; Amirat, Y. Control of Upper-Limb Power-Assist Exoskeleton Using A Human-Robot Interface Based on Motion Intention Recognition. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 1257–1270. [CrossRef]
23. Song, S.; Li, Z.; Meng, Q.H.; Yu, H.; Ren, H. Real-Time Shape Estimation for Wire-Driven Flexible Robots With Multiple Bending Sections Based on Quadratic Bézier Curves. *IEEE Sensors J.* **2015**, *15*, 6326–6334. [CrossRef]
24. Croom, J.M.; Rucker, D.C.; Romano, J.M.; Iii, R.J.W. Visual sensing of continuum robot shape using self-organizing maps. In Proceedings of the IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010.
25. Xu, W.; Foong, R.P.L.; Ren, H. Maker based shape tracking of a flexible serpentine manipulator. In Proceedings of the IEEE International Conference on Information and Automation, Lijiang, China, 8–10 August 2015.
26. Xu, R.; Yurkewich, A.; Patel, R.V. Curvature, Torsion, and Force Sensing in Continuum Robots Using Helically Wrapped FBG Sensors. *IEEE Robot. Autom. Lett.* **2016**, *1*, 1052–1059. [CrossRef]
27. Qian, S.; Zheng, T.; Shen, W.; Bai, H. Feedback control of Cable-Driven continuum robot based on flexible perception. In Proceedings of the International Symposium on Control Engineering and Robotics, Hangzhou, China, 17–19 February 2023; pp. 64–69.
28. Shen, W.; Yang, G.; Wang, Y.; Fang, Z.; Li, F.; Wang, H. Design and performance analysis of a Cable-Driven continuum robotic joint module. In Proceedings of the International Symposium on Robotics and Intelligent Manufacturing Technology, Changzhou, China, 24–26 September 2021; pp. 427–433.

sensors

MDPI

Article

Sensory–Motor Loop Adaptation in Boolean Network Robots

Michele Braccini [1,*], Yuri Gardinazzi [2,3,4], Andrea Roli [1,5] and Marco Villani [4,5]

1 Department of Computer Science and Engineering, University of Bologna, 47521 Cesena, Italy; andrea.roli@unibo.it
2 Department of Mathematics, Informatics and Geosciences, University of Trieste, 34127 Trieste, Italy; yuri.gardinazzi@phd.units.it
3 AREA Science Park, 34149 Trieste, Italy
4 Department of Physics, Informatics and Mathematics, University of Modena and Reggio Emilia, 41125 Modena, Italy; marco.villani@unimore.it
5 European Centre for Living Technology, 30123 Venice, Italy
* Correspondence: m.braccini@unibo.it

Abstract: Recent technological advances have made it possible to produce tiny robots equipped with simple sensors and effectors. Micro-robots are particularly suitable for scenarios such as exploration of hostile environments, and emergency intervention, e.g., in areas subject to earthquakes or fires. A crucial desirable feature of such a robot is the capability of adapting to the specific environment in which it has to operate. Given the limited computational capabilities of a micro-robot, this property cannot be achieved by complicated software but it rather should come from the flexibility of simple control mechanisms, such as the sensory–motor loop. In this work, we explore the possibility of equipping simple robots controlled by Boolean networks with the capability of modulating their sensory–motor loop such that their behavior adapts to the incumbent environmental conditions. This study builds upon the cybernetic concept of homeostasis, which is the property of maintaining essential parameters inside vital ranges, and analyzes the performance of adaptive mechanisms intervening in the sensory–motor loop. In particular, we focus on the possibility of maneuvering the robot's effectors such that both their connections to network nodes and environmental features can be adapted. As the actions the robot takes have a feedback effect to its sensors mediated by the environment, this mechanism makes it possible to tune the sensory–motor loop, which, in turn, determines the robot's behavior. We study this general setting in simulation and assess to what extent this mechanism can sustain the homeostasis of the robot. Our results show that controllers made of random Boolean networks in critical and chaotic regimes can be tuned such that their homeostasis in different environments is kept. This outcome is a step towards the design and deployment of controllers for micro-robots able to adapt to different environments.

Keywords: sensory-motor loop; Boolean networks; homeostasis

Citation: Braccini, M.; Gardinazzi, Y.; Roli, A.; Villani, M. Sensory–Motor Loop Adaptation in Boolean Network Robots. *Sensors* **2024**, *24*, 3393. https://doi.org/10.3390/s24113393

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 24 April 2024
Revised: 20 May 2024
Accepted: 21 May 2024
Published: 24 May 2024

1. Introduction

The pace at which robot technology proceeds has rapidly stepped up in recent years. In addition to complex robots, such as humanoid ones and unmanned rovers, small robots of millimeter or micrometer size have been released. Such robots are often inspired by insects and bugs, and are usually equipped with a few simple sensors and actuators, and are characterized by limited computational capabilities. A prominent scenario for these robots is that of swarms to be applied in exploration of hostile environments and emergency intervention. The missions robots have to accomplish in these situations require capabilities such as autonomy and adaptiveness. AI techniques provide effective and efficient solutions to address these issues. Nevertheless, the reduced computational capabilities of micro-robots make these techniques often inapplicable, hence the need for alternative forms of control that are simpler yet still able to produce adaptive behaviors. A theoretical framework for defining this problem and finding solutions to it is that of embodied AI [1,2], which relies on the sensory–motor loop that connects the robot with the environment and makes it possible for a robot to adapt. The sensors produce the information the robot uses to determine its actions, which modify the environment through the effectors; this, in turn, generates possibly new sensor readings. Hence, the actions of the robot can influence its sensing. This feedback loop is at the basis of elementary yet non-trivial behaviors, such as gradient following and motion avoiding dangerous areas. The sensory–motor loop also plays a significant role in the integration in time of sensor signals, often making it possible to extract relevant information from the environment that is not directly perceived by sensors. For example, by combining movement and light perception, a robot can estimate the height of an object [3]. However, for a robot to be adaptive, mechanisms to change and tune the sensory–motor loop are needed. As beautifully stated by Ashby [4], organisms adapt to their environment "for the better". In robotics, this condition is usually expressed in terms of a merit factor expressed in terms of a utility function, which guides the adaptation process. The utility function is mostly based on the specific task the robot has to accomplish and it is given by the designer beforehand. This is of course a common and feasible approach when the most relevant features of the environment and the task can be modeled in advance. Nevertheless, in some situations, like emergencies, such predefined utility functions might not capture the actual "good" for the robot. In such cases, the robots should be equipped with general, task-agnostic rules for acting and adapting. These rules are commonly called values. The ultimate value is self-maintaining, which can be formally stated as homeostasis, i.e., the capability of keeping the system in a working condition despite the changes in the environment. Notably, the ability of maintaining homeostasis in changing environments can be interpreted as a way of making sense of the environmental features that are relevant to the robot. This perspective is not new, as it has been thoroughly discussed in fields such as cybernetics and cognitive systems [3–13]. Keeping this context as a reference, in this work, we discuss the results of a study aimed at investigating and exploring possible adaptive mechanisms for micro-robots controlled by simple circuits. Since the typologies and features of micro-robots are—and will be—extremely varied, we focus on a kind of control software that can match any specific instance of physical robots. Therefore, the level of abstraction of our investigation is at the control layer, independently of the actual physical robot. In particular, we concentrate our attention to control software based on Boolean networks and we study the properties of simulated robots that have to keep their homeostasis in front of environment changes. This work is aimed at providing guidelines and suggestions for designing new control mechanisms capable of enabling adaptivity, especially in micro-robots. Other authors have approached the topic of homeostasis from different perspectives, e.g., through the lens of active inference [14] and within the framework of active perception [15], to name just a few.

Our investigation is inspired by cybernetics, minimally cognitive robotics and dynamical systems, and tries to bridge this theoretical background to robot control methods and technologies. To the best of our knowledge, this is the first time such a study is presented. Nevertheless, Boolean network controllers for robots—and their connections to dynamical criticality—have been already studied, mainly by some authors of this contribution, [16,17]. In [17], an offline adaptation of a robot controlled by Boolean networks has been presented, while online adaptation has been investigated in [16], where it is shown that Boolean networks poised at a dynamical regime between order and disorder are the one that enable the robot to achieve the best performance. The adaptive mechanisms devised in [16] are similar to those used in this work. Nevertheless, in previous works, robots learned to accomplish a specific task in a given environment, while they have to be able to adapt and maintain their operative status in this work.

In Section 1, we introduce the model used in our experiments. As we are dealing with simulated robots, we will use the terms agent and robot interchangeably. In Section 2, we present the robot's and environment's models along with the adaptive mechanism employed by the robot to achieve homeostatic condition when coping with a changed environment. The results for the specific scenario of coupled robot–environment systems studied are described in Section 3, while Section 4 discusses their possible general implications. Section 5 is devoted to the presentation of the conclusion.

2. The Model

The main elements that have to be defined for setting up experiments of autonomous agents in a changing environment are (i) an autonomous agent, and (ii) a dynamic environment in which the moves of an agent can have effects in a simple (enough to be useful for designing experiments) but effective way. Further, (iii) it is necessary to allow the agent to establish what is advantageous and what is not.

This last characteristic is perhaps the most delicate, as we aim at meanings/benefits to emerge without design, or to be "autonomously decided" by the system immersed in a dynamically changing environment, without an explicit design by the experimenters. At the same time, much of the meaning/usefulness actually depends on the interaction between the agent and the particular environment and on the active feedback loops. The role of the sensory–motor mechanisms lies precisely in this last aspect, which we will focus on in this work. In order to highlight this last aspect, we will assume that the agent is already in a "good" situation due to a previous history (evolution for living beings, explicit design for artificial systems), and it is exposed to new environments. A general goal of any agent in such a situation is to keep its internal situation stable and coherent (homeostasis). If the new environment has changed this situation, the agent should react by influencing the environment in order to restore the state of well-being.

2.1. Agent's Model

We are interested in simulating an agent moving in an unknown environment: the agent and the environment each have their own dynamics, while the exchange of information between the two systems occurs through the sensors and actuators of the agent. The internal states of the agent and their relationships are represented through a random Boolean network (RBN). The environment must have non-obvious dynamics, which can be modified through the interaction of an external system (the robot) with the features of the environment itself. An elegant way to model a reactive environment is to realize it via another RBN.

Boolean networks (BNs) are simple but powerful discrete models of gene regulatory networks, introduced by Kauffman [18,19]. BNs can be defined mathematically as a directed graph with N nodes whose node states are described by Boolean variables x_i , $i = 1, \dots, N$, which can take the values ON or OFF, and their evolution over time is described by Boolean functions $f_i\left(x_{i_1}, \dots, x_{i_{Mi}}\right)$, where Mi is the number of inputs of node i. Random Boolean networks (RBNs) are a special class of BNs, a class determined by parameters that impose constraints on the connections between nodes and on their dynamics.

A noteworthy RBN class is one in which each node receives an exact number of inputs, determined by the K parameter, chosen randomly from the other nodes and avoiding self-loops, and in which the probability of having the value 1 in each entry in the truth tables of the Boolean functions is determined by the p parameters, called bias. In these works [20,21], the authors showed that this mathematical relationship between p and K, $K = [2p\,(1-p)]^{-1}$, describes the set of RBNs that fall in the order-disorder phase transition, the so-called critical line for RBNs. Perturbations in RBNs in the ordered dynamical regime generally die out quickly, whereas they spread out in disordered networks. Networks in disorder regime are also characterized by very long cyclic attractors.

In contrast, networks in the critical regime—those falling into the region described by the above equation—show a balance between adaptability and robustness that has led to the formulation of the conjecture that life and computation exist at the edge of chaos, the criticality hypothesis [22–25].

Critical RBNs have proven capable of properly reproducing many biological processes [26–34] and have proven efficient when used as controllers of artificial agents, such as robots [27].

The robot's effectors can therefore be represented through nodes that are able to influence the state of nodes belonging to the environment (hereinafter called "features"), while its sensors are nodes that receive signals from some features of the environment. For simplicity, we assume that the value of a sensor node is influenced by only one feature of the environment, and similarly the value of an effector node influences only one feature of the environment. In a physical implementation, this implies that the Boolean value of each output node of the network controls an effector via a given encoding (e.g., in the case of a device controlled by an impulse, the Boolean values naturally correspond to minimum/maximum voltage applied). Similarly, an input node assumes a Boolean value which is the result of an encoding from a possibly continuous to Boolean domain (e.g., values from [0, 1] can be converted simply by introducing a threshold). The state of the robot's well-being is measured by the distance from homeostasis of a subset of its nodes (the "essential nodes"). To avoid trivial dynamics, the three sets (sensor nodes, effector nodes and essential nodes), whose sum of cardinalities is less than the total number of robot nodes, are non-overlapping (see Figure 1).

There is a "universal time", common to the robot and the environment, which proceeds in discrete steps and regulates the succession of events. However, the internal processes of the robot can be slower or faster than the environmental processes: each of the two systems therefore has an internal variable (called, respectively, agent_step and env_step) which establishes how many "universal" steps must pass to allow the application of the internal dynamical rules. If the values of these variables are the same both systems proceed in synchronization, otherwise the system that has a higher value is "slower" than the other one; as a special case, if the value of both the variables is equal to 1, robot and environment proceed paired, at maximum speed.

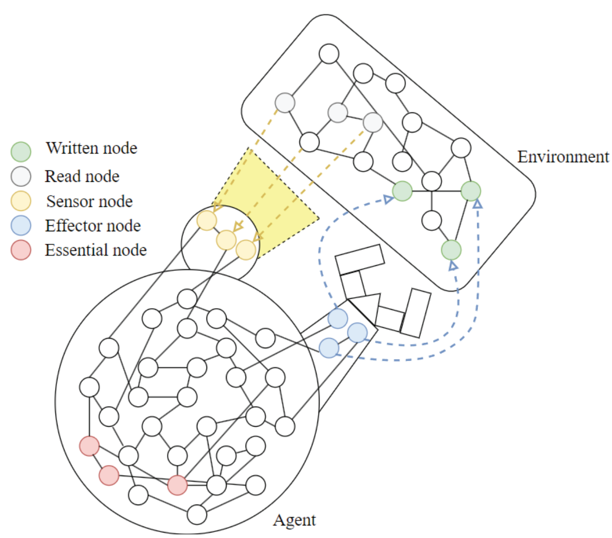


Figure 1. Agent interacting with an environment. It detects the environment features by using its sensors and acts on some features of the environment by using the effector nodes. The essential nodes are highlighted in pink.

It is therefore needed to describe the interaction between two systems that can have different speeds. For each sensor node, it is necessary to define (i) the perceived value of the environmental feature to which it is associated and (ii) its response to that value. It was therefore decided that (a) the perceived value is equal to the average of a number of universal steps of the environment equal to agent_step, while (b) the response of the sensor node is equal to "0" or "1" depending on that the perceived average is below or above a threshold $\theta \in]0{,}1[$ (in the particular case that *env_step* is equal to 1, the perceived value is the value of the node itself, and any threshold does not modify its value).

An effector node acts on a specific feature of the environment. The mechanism is symmetrical to the previous one: (a) the value perceived by the environmental feature is equal to the average of a number of universal steps of the robot equal to env_step, while (b) the response of the feature is equal to "0" or "1" depending on that the perceived average is below or above a threshold $\theta \in]0{,}1[$. This "response" is the value that the node effector of the robot manages to apply on the feature of the environment. See Figure 2.

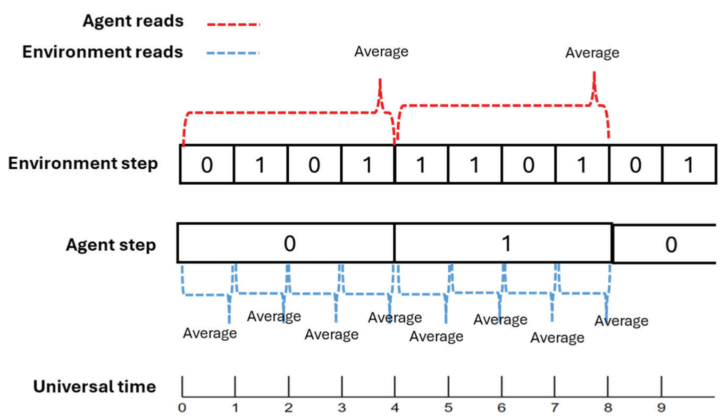


Figure 2. Example of how nodes are read with an environment with env_step = 1 and agent with agent_step = 4. In this case, the environment just averages on one value of the agent, while the agent averages the 4 last environment steps.

As anticipated, we assume that the robot without an environment is already in its ideal situation: once exposed to an unknown environment it must therefore act in such a way as to restore this situation, at least for the essential nodes. The ideal profile is therefore defined as the average, essential node by essential node, taken over a η steps performed by the robot without any environment (be $\underline{X}_{id,\eta}$ the vector of these averages).

We then evaluate the distance from this ideal situation during the life of an agent T times (each evaluation being interspersed with ω agent steps) and consider the average of these values; in this paper we assume the Euclidean distance:

$$D = \frac{\sum_{n=1}^{T} d\left(\underline{X}_{e,\omega}, \underline{X}_{id,\eta}\right)}{T} = \frac{\sum_{n=1}^{T} \left\|\underline{X}_{e,\omega}, \underline{X}_{id,\eta}\right\|_2}{T} \quad (1)$$

The lifetime of an agent measured in universal steps is therefore equal to T "trials", multiplied by ω steps, multiplied by *agent_step*:

$$L_{agent} = \omega \cdot T \cdot agent\_step \quad (2)$$

2.2. Adaptive Mechanisms

To use the characteristics of the environment in order to maintain its well-being (homeostasis), the robot must therefore choose the features on which to act and those from which to derive utility.

To avoid trivial situations in this paper we do not allow the robot to choose the features of the environment from which to draw utility, nor its nodes that are connected to these features. The aim of the robot is therefore given its characteristics (its internal structure and the characteristics of the sensor nodes), to discover where and how to act on the environment (through the choice of the effector nodes and the features of the environment on which they act), in order to receive from it (through the features of the environment to which the sensor nodes are connected) the stimuli needed to maintain homeostasis (minimize the distance of the essential nodes from their ideal profiles).

We can think of the robot's learning as a succession of its versions, each deriving from the best of the previous ones. Each version has the same lifespan and starts from the same initial conditions and is evaluated in the way just described (Equation (1)). The robot thus tests in an organized manner the various possibilities offered by the interaction between itself and the environment, until it reaches the maximum possible number of tests. In each change step, the next version is accepted only if its distance from the ideal profiles of the essential nodes is equal to or lower than that of the previous one: at the end of the process the final version of the robot is obtained.

Changes can be performed following three strategies:

1. Change the effector nodes, leaving the features of the environment on which they act unchanged;
2. Maintain the effector nodes, and change the features of the environment on which they act;
3. Change both the effector nodes and the features of the environment on which they act.

The basic steps defining adaptation strategy 1 and 3 are actually the same as those used in the online adaptation mechanism proposed in [27]. See Figure 3.

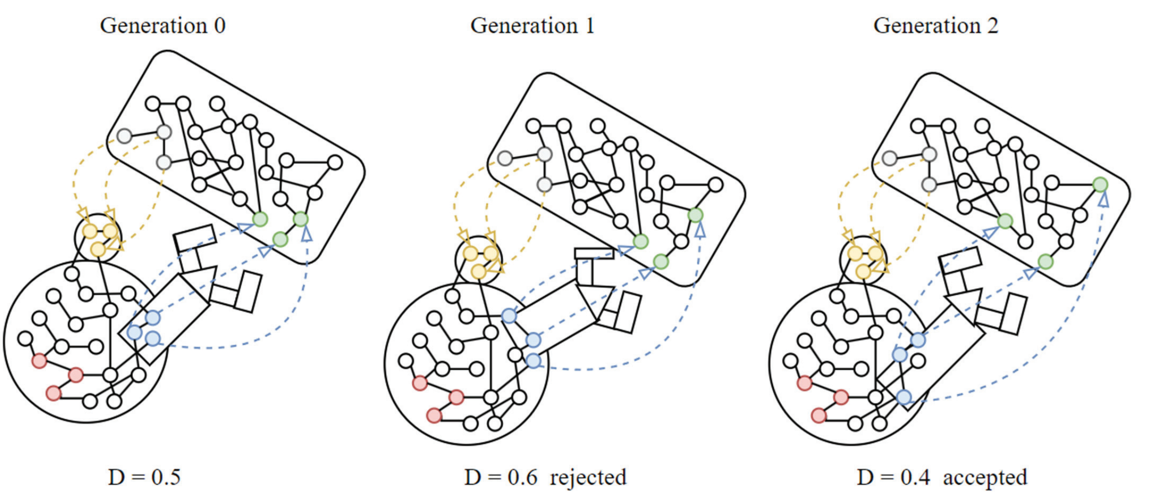


Figure 3. Example of evolution of an agent. The generation 1 is rejected because a higher value of D.

3. Results

Below, we will present results for the case that, roughly speaking, we can describe as "fast environment and slow robot", that is, the case where *env_step* = 1 and *agent_step* = 10.

In general, this scenario should give the dynamics of the environment time to relax on one of its attractors upon the perturbation received by the agent on its features.

This should, in turn, allow the robot to receive a response (sensory input) mediated by the environment that is as time consistent as possible with the action that triggered it (performed by the robot itself), thus unaffected by possible fluctuations resulting from transients of the environment dynamics.

We are also interested in observing how the dynamical regime expressed by a Boolean network can influence its bouquet of responses if immersed in unknown environments.

We calculate Derrida's coefficient to evaluate the dynamical regime in which a Boolean network operates. Derrida's coefficient λ measures the average level of propagation of a perturbation after a simulation step. Statistically, $\lambda > 1$ characterizes chaotic networks, $\lambda < 1$ ordered networks while $\lambda = 1$ critical networks. To calculate it for a single Boolean state, we copied or negated one of its variables; we then performed a synchronous update for both the original and the perturbed state and finally measured the Hamming distance between the two resulting states. Next, we averaged the values by repeating this procedure 80 times for each of the 250 initial conditions, resulting in 20,000 total occurrences.

3.1. The Three Strategies

We carried out a series of experiments, in which robots were immersed in unknown environments. These environments must have their own dynamics and timing: for this purpose we once again used the RBN framework. We, therefore, created 10 instances of RBNs, each having 50 nodes and average connectivity equal to 3: the bias p equal to 0.21 implies a critical dynamical regime (as a check, we measured the Derrida coefficients of each environment, which all turned out to be close to 1.0).

In a preliminary series of experiments, we evaluated the homeostasis recovery capacity of ensembles of robots (with critical and chaotic dynamical regimes) once they were exposed to each of 10 environments. Once immersed in an environment, each robot was typically distanced from its "ideal" state, and applied one of the three strategies reported above to decrease (if not eliminate completely) this distance (measured as in Equation (1)). In each environment E_j, we used the relative decrease in the distance from the ideal profile as an evaluation of the capacity for homeostasis of robot R_i:

$$G_i(E_j) = \frac{D^i_{initial}(E_j) - D^i_{final}(E_j)}{D^i_{initial}(E_j)} \tag{3}$$

so that the performance of each robot R_i is the average relative decrement:

$$P_i = \frac{1}{10}\sum_{j=1}^{10} G_i(E_j) \tag{4}$$

In the following, we will use robots having an internal system schematized by using RBNs composed of 100 nodes, of which 3 sensor nodes, 3 effector nodes and 3 effective nodes. The three subsets are non-overlapping, and direct connections (i) between sensor nodes and effector nodes, (ii) between sensor nodes and effective nodes, and (iii) between effective nodes and effector nodes are excluded.

Statistics using ensembles of 50 robots have indicated that the third strategy is the one that allows for better versatility (see Table 1): in what follows we will therefore focus on this modality.

Table 1. The average performances of ensembles (whose cardinality is equal to 40) of critical and chaotic robots when using the three adaptation strategies. The average is made on the relative decrease (Formula (3)). Strategy 3, which allows changes both in the choice of effector nodes and in the environmental features touched, has better results than the other two strategies.

	Average Performance		
	Ordered RBNs	**Critical RBNs**	**Chaotic RBNs**
STRAT. 1	0	0.075	0.196
STRAT. 2	0	0.131	0.264
STRAT. 3	0	0.194	0.308

In these experiments, as in the following ones, the difference between the robots of the ordered, critical and chaotic sets lies—without loss of generality—in the Boolean functions used. Each created topology was then combined with sets of Boolean functions having biases equal to 0.1, 0.21 and 0.5: the common connectivity $k = 3$ leads these RBNs, if isolated, to exhibit, respectively, ordered, critical or chaotic behaviors.

The ordered ensembles have always shown zero average performance values: we will therefore avoid dealing with them. In Table 1, the data seem to indicate a better performance than chaotic RBNs: to interpret the result, however, a more detailed analysis is necessary (see Section 3.3). During the research, we tested different sets of parameters: in the simulations presented in this work, we used the values indicated in Table 2.

Table 2. Values of the main parameters used during the simulations in this work. The length of each simulation is equal to omega*trial_number*trial_step = 280,000 "universal steps". The probability that an effector node is modified (prob_change_effector) or of modifying the choice of a feature of the environment to be influenced (prob_change_feature) is equal to 0.0 or 0.5 depending on the strategy chosen; the sensor nodes and the environmental features they monitor do not change.

Variable	**Value**	**Variable**	**Value**
trial_number	2800	Number of nodes (agent)	100
trial_step	20	Number of nodes (environment)	50
w	5	Average connectivity k	3
prob_change_sensor	0;	Bias: Ordered ensemble	0.1
prob_change_effector	0; 0.5	Bias: Critical ensemble	0.21
prob_change_feature	0; 0.5	Bias: Disordered ensemble	0.5

3.2. A Common Behavior

An interesting characteristic, common to all adaptation strategies, concerns the type of action that agents can exercise once adapted to the new environment. Agents typically change both the features of the environment influenced by them (when they can, and therefore with the exception of strategy (1) and their effector nodes: the action conducted by these nodes does not, however, always have the same effects. In fact, a constant action exercised on particular areas of the environment may have the effect of an oscillating signal on the sensor nodes (the critical agent C1 in Figure 4), while an oscillating action on other areas of the environment causes a different signal oscillating on the sensor nodes (the critical agent C2 in Figure 5 or the disordered agent D1 in Figure 6). In these cases, the need for a final oscillating situation of the essential nodes depends on the fact that the three agents all require average node activities different from stable values equal to 0 or 1, but the notable point is that this situation can be obtained either by exercising a constant action and an oscillating one.

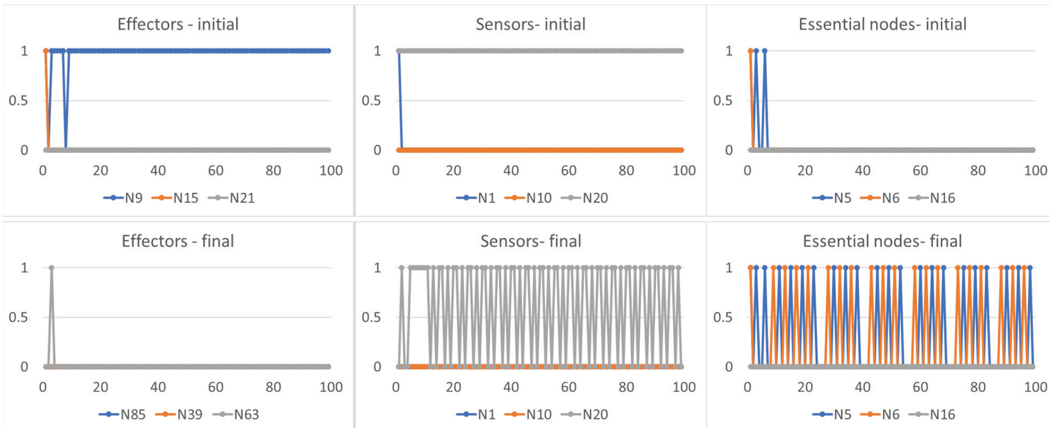


Figure 4. The first row shows the trajectory of the effector, sensor and essential nodes of the critical agent C1 when it is exposed to one of the environments. The second row reports the trajectory of the effector nodes (in the case the agent has modified its effector set, different from the previous ones), sensor and essential nodes of the same agent once it has adapted to the new environment. The features of the environment on which the agent exercises its final action can be different from those on which it initially acted. It is possible to note that a constant action on the environment (first plot of the second row) leads to an oscillating signal in the sensor nodes (second plot of the second row): this action brought the average activity of the essential nodes (third plot of the second row) close to the homeostatic average, in this case equal to [0.2, 0.2, 0.0].

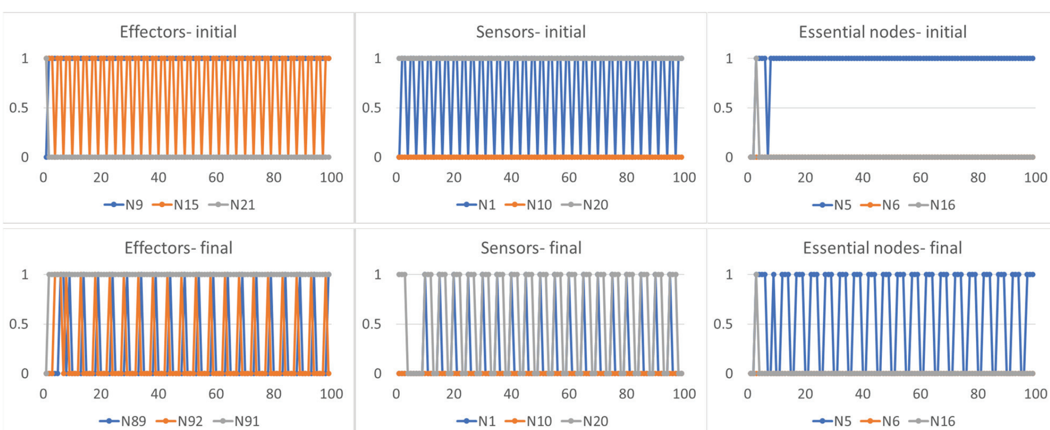


Figure 5. The first row shows the trajectory of the effector, sensor and essential nodes of the critical agent C2 when it is exposed to one of the environments. As in Figure 4, the second row reports the trajectory of the effector, sensor and essential nodes of the same agent once it has adapted to the new environment. It is possible to note that a different oscillating action on the environment (second row, first plot) leads to an oscillating signal in the sensor nodes (second row, second plot) able to induce on the essential nodes (second row, third plot) an average activity the close to the homeostatic average [0.2, 0.0, 0.8].

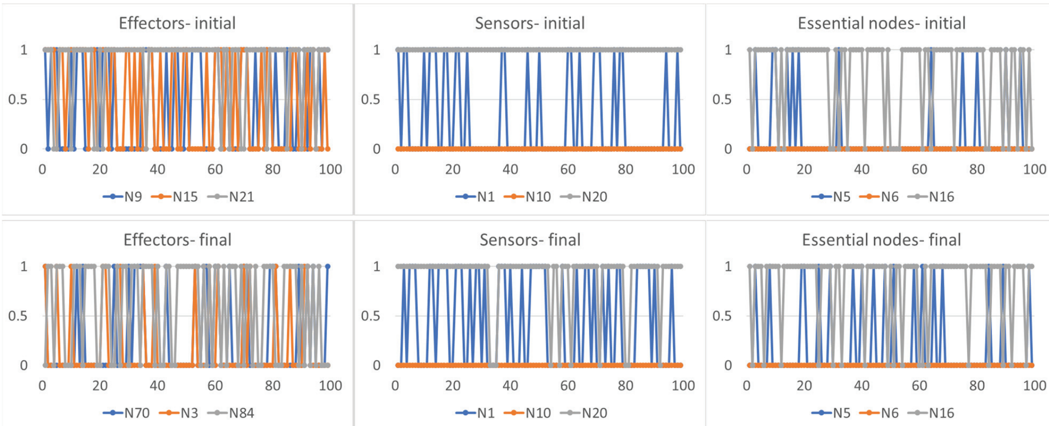


Figure 6. The first row shows the trajectory of the effector, sensor and essential nodes of the disordered agent D1 when it is exposed to one of the environments. As in Figure 4, the second row reports the trajectory of the effector, sensor and essential nodes of the same agent once it has adapted to the new environment. It is possible to note that a different oscillating action on the environment (second row, first plot) leads to an oscillating signal in the sensor nodes (second row, second plot) able to induce on the essential nodes (second row, third plot) an average activity the close to the homeostatic average [0.6, 0.0, 0.1].

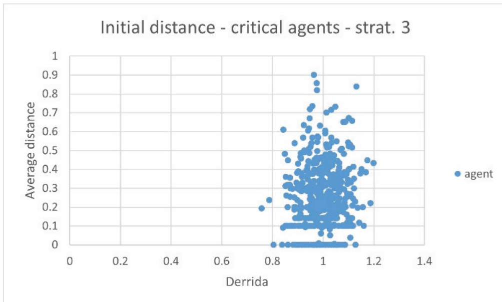
The agents therefore used the environment to which they were exposed in a non-trivial way, including its dynamic characteristics in their effector–sensor loop to achieve homeostasis. Remarkably, the methods used are such as to make this action difficult to identify based on internal observations alone: a similar action (the induction of particular oscillations in the essential nodes) was achieved in one case through a constant action on the environment (agent C1) and in other cases through an oscillating action (agents C2 and D1).

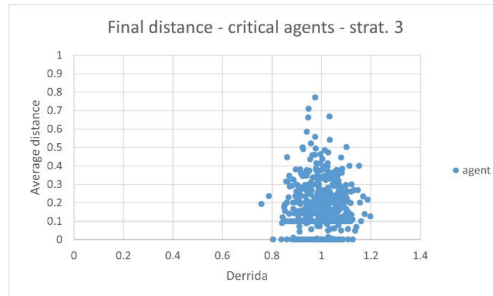
The scheme we adopted therefore allows us to identify the dynamic role of the coupling between the environment and the agent, highlighting the explicit action of the agent of incorporation into the effector–sensor loop of the dynamics of the environment for the purposes of its own well-being.

3.3. The Third Strategy

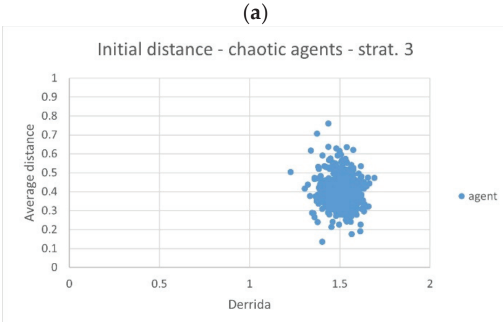
We therefore focused on the third strategy, using larger ensembles (each ensemble being composed of 500 RBNs). It is possible to make some interesting considerations.

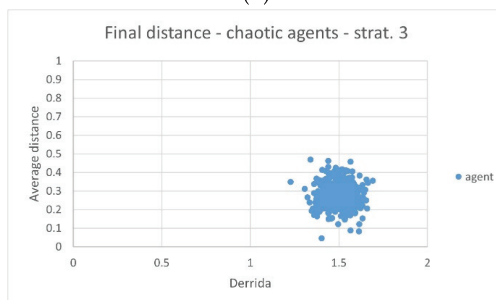
A first observation concerns the fact that the critical ensemble shows a great variety of behaviors, from RBNs that have not received any perturbation from the environment, to RBNs that have been strongly perturbed, in a way that is little influenced by the Derrida coefficient, consistently always placed around the value 1.0 (Figure 7a). The chaotic ensemble shows much greater uniformity (Figure 7c).





(**a**) (**b**)





(**c**) (**d**)

Figure 7. (**a**) The initial distance from the ideal profile of the 500 critical RBNs as soon as they are exposed in each of the 10 environments (each point corresponds to the average distance value over the 10 environments). (**b**) The average distance from the ideal profile of the 500 critical RBNs once the adaptation has been performed following strategy 3. (**c**) The average initial distance from the ideal profile of the 500 chaotic RBNs. (**d**) The average final distance from the ideal profile of the 500 chaotic RBNs once the adaptation has been performed following strategy 3. In all graphs, the X axis indicates the Derrida coefficient of each individual RBN.

Both ensembles show improvements once the RBNs are allowed to adapt to different environments (Figure 7b,d). Again, the RBNs belonging to the chaotic ensemble show a uniformity of behavior that seems absent from the critical ensemble.

This diversity of behavior allows us to make various considerations. The chaotic regime (better to say "extremely disordered") leads to a uniformity of behavior, little dependent on the details of the individual instances. All RBNs are perturbed by external environments, without exception; all RBNs show traces of improvement once adaptation is allowed. The RBNs belonging to the critical ensemble show extreme diversity: many are not perturbed by the environments to which they have been exposed, many have been highly perturbed, more than the most perturbed chaotic RBN. Many of the perturbed RBNs show improvements. From an evolutionary point of view, this greater diversity of behavior is an advantage, allowing the subsequent selection of the variant most suited to the need. Table 3 shows how many of the critical RBNs were not perturbed by the new environments, and how many, even if initially perturbed, did not improve.

Table 3. The table shows in the first row the number of critical RBNs that have been improved by adaptation phase, i.e., whose final distance from the homeostatic condition is less than the initial distance. The second row shows the networks that have been affected by the change in environment, but have not been able to move closer to their homeostatic condition, i.e., for the network i it occurs that $D^i_{final}\left(E_j\right) \geq D^i_{initial}\left(E_j\right)$. Finally, the last row shows the networks whose homeostasis has not been perturbed by environmental change.

# RBNs	Critical RBNs
Improved	266
Did not improve but perturbed	173
Not perturbed	61

It is interesting to observe what happens to critical RBNs that have been perturbed by exposure to unknown environments and improved once they were allowed to adapt (a situation similar to that of the RBNs belonging to the chaotic ensemble).

In this case, it is possible to see that there are critical RBNs capable of completely zeroing out the distance from their ideal profiles, and that there are RBNs that, despite starting from a very disadvantaged position, are able to express notable recoveries (the blurring of the colors in Figure 8b, in which the yellow dots and gray dots show even greater improvements than orange and blue dots).

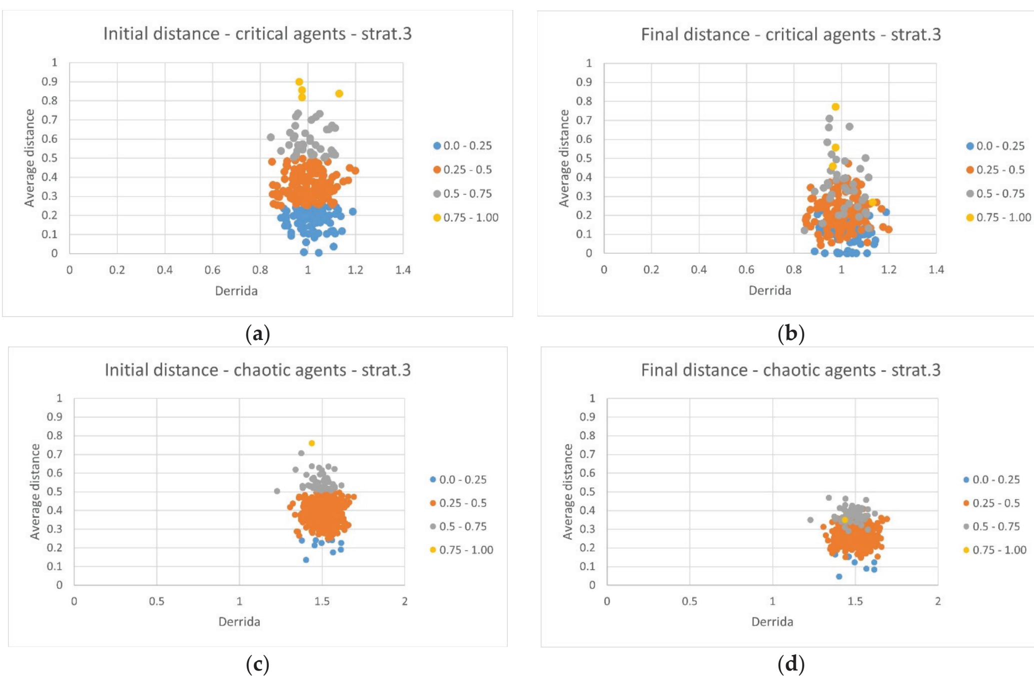


Figure 8. (**a**) The average initial distance from the ideal profile of the critical RBNs that were perturbed and then improved through the application of strategy 3, highlighted in different colors depending on the initial distance. (**b**) The average distance from the ideal profile once the adaptation has been performed following strategy 3. (**c**) The average initial distance from the ideal profile of the chaotic RBNs. (**d**) The average final distance from the ideal profile of the chaotic RBNs once the adaptation has been performed following strategy 3. In all graphs, the colors identify 4 bands depending on the distance: blue for the interval [0.0–0.25[, orange for the interval [0.25–0.50[, gray for [0.50–0.75[and yellow for [0.75–1.00].

Similar capabilities are not shown by the RBNs belonging to the chaotic ensemble, which maintain uniformity of behavior (the stable division into separate colored bands). Furthermore, in this situation, the RBNs belonging to the critical ensemble also show an average recovery capacity higher than that of the RBNs belonging to the chaotic ensemble (Table 4).

Table 4. The average performances of the ensembles of critical and chaotic robots (whose cardinality is 500 for chaotic RBNs and 266 for critical RBNs that showed improvement) when using the third adaptation strategy, i.e., the best performing one (see Table 1).

	Critical RBNs (Only the Improved Ones)	Chaotic RBNs
MIN	0.001	0.115
MAX	1.000	0.658
AVERAGE	0.382	0.310
MEDIAN	0.377	0.300

In other words, if we extract from the critical ensemble the RBNs that are sensitive to the environment and capable of improvement, we obtain systems that are typically better than the corresponding systems showing a chaotic dynamical regime. The results therefore agree with Kauffman's hypotheses, which imagines a particular role for critical dynamical regimes, which (i) show greater initial variability are susceptible to selection and (ii) are capable of supporting a high degree of adaptability. On the other hand, the uniformity of behavior of the chaotic ensemble prevents the possibility of having particularly high-performance RBNs.

4. Discussion

The model chosen for our experiments, although simple, makes it possible to address fundamental questions concerning robot adaptation. Rephrasing Ashby [4] in current terms, we have explored possible ways for achieving adaptation ("identify the nature of the change which shows learning") and investigated the circumstances that make this adaptation more efficient ("to find why such changes should tend to cause better adaptation for the whole organism").

The use of RBNs to control the agents on the one hand provides generality to the results we observed—indeed, only the dynamical regime matters—and, on the other hand, it shows that network-based controllers can be subject to adaptive mechanisms that confer non-trivial adaptive capabilities to robots. Furthermore, the choice of homeostasis as a merit factor for guiding adaptation addresses the issue of providing robots with autonomy: once a robot is capable of changing some of its control parameters such that its survival is preserved, then other task-specific utility functions can be optimized. This capability may well constitute a basic competence for a layer in a subsumption architecture [35].

A prudent reader might observe that a limitation of our results is that they have been attained in simulation only. Tests on real robots are indeed required to thoroughly assess the robustness of the phenomena we observed. However, previous works with Boolean network controlled robots [17] have validated on real robots (i.e., e-puck robots [36]) the results obtained in simulation. Furthermore, the adaptive process that we used is analogous to an adaptive walk in a configuration space, which can be considered as a special case of evolutionary algorithms used in robotics. Many results in evolutionary robotics and automatic design of control software for robots have been validated on real robots, for example [37–42]. Therefore, we are rather confident that the core behavior observed can be also observed when the controller we have studied is implemented on physical robots.

Central in our model is the adaptation of the sensory–motor loop, which is at the basis of behavior-based robotics. Here, we suppose that the sensors of the robot are given and not subject to adjustments, whereas the actuators are those devices that can be adapted to the environment in such a way that the robot can keep its homeostasis. As a side comment, we observe that this setting loosely resembles approaches based on the free energy principle [43], in which actions are taken such that the free energy of the system is minimized. Having only the actuators that can undergo changes clearly shows whether the agent is actually able to choose suitable actions to adapt: when the agent is placed in a new environment and its homeostasis is negatively perturbed, only by means of proper actions it can influence the environment to produce good and valuable inputs to its sensors. This has two main implications. The first one is that robots capable of adapting successfully exploit the features of the new environment to which they are exposed. In other terms, the anomalies produced by the environment trigger new effective behaviors in the robots. Therefore, effective behavior control in robots can profit from the combination of some degree of adaptation with tests in differing environments. The second important consequence is that the new action-sensor feedback, created by adaptation, that restores homeostasis in the agent identifies a sense-making property: the agent selects those features of the environment that matters to its survival and learns how to act on them under the conditions posited by the sensors and the internal mechanisms of the agent.

Finally, our results provide support to the conjecture that critical regimes are the most suited for adaptation and evolution. Another proof of the advantage of critical networks as controllers in Boolean network robots was already observed in the work [16]. The cited work analyzed the performance of an online adaptation scheme very similar to the one used in this paper, but employing real robots in obstacle avoidance and foraging tasks. However, the results reported in this work are more general, in that the Boolean network representing the agent does not attempt to improve a task-specific figure of merit, but instead tries to adapt to achieve its homeostatic condition.

This result can be generalized to other network-based controllers, such as nanowire networks [44].

5. Conclusions

In this work, we have investigated how robots controlled by Boolean networks can adjust their sensory–motor loop to adapt to environment changes, such that their homeostasis is restored. Some general implications of the results we obtained have been discussed in the previous section. However, we would like to point out that this work, through the proposed conceptual and experimental framework, paves the way for the discovery of the general principles that enable agent homeostasis in scenarios characterized by changing environments. The importance of the criticality of the agent's controller adds to Ashby's "Principle of Requisite Variety" [45,46] and begins to outline the framework of a truly homeostatic robot, a crucial first step toward robot autonomy and possibly to machines endowed with feeling [47]. A possible roadmap of future work involves testing this agent–environment framework (i) with real robots endowed with a physical body (ii) with the ability to adapt both its robotic morphology and its controller; (iii) with changing environments (ecological niches); (iv) in coevolutionary settings (i.e., both the agent and environment can change in response of the other previous and current perceivable actions); and finally (v) in multi-robot scenarios. In addition, our efforts will be devoted to characterize the processing of information flow that homeostasis induces between sensors, controllers, actuators, and the environment, which can be analyzed with Information Theory measures.

We are currently running experiments with larger networks to assess the results concerning the dynamical regimes. In addition, experiments with different relative update frequencies between agents and environment are under study. In addition to these further analyses, feasibility studies concerning the implementation on physical robots of the mechanisms we have studied in simulation are planned. Micro-robots can now be cheaply produced and it is plausible that they can be equipped with network-based controllers. In these robots, the essential variables might be represented by energy and efficiency parameters; therefore, basic capabilities of keeping these variables within working ranges would be of extreme utility. This survival competence can then provide the basis for more advanced and special purpose behaviors. Finally, the experimental setting we have designed makes it possible to investigate the relation between adaptation and high-level principles, such as the free energy principle and autopoiesis.

Author Contributions: Conceptualization, M.B., A.R. and M.V.; Methodology, M.B., Y.G., A.R. and M.V.; Software, Y.G.; Investigation, M.B., Y.G., A.R. and M.V.; Data curation, Y.G.; Writing—original draft, M.B., Y.G., A.R. and M.V.; Writing—review & editing, M.B., Y.G., A.R. and M.V.; Supervision, M.B., A.R. and M.V. All authors have read and agreed to the published version of the manuscript.

Funding: A.R. acknowledges support by the PRIN 2022 research project of the Italian Ministry of University and Research titled Org(SB-EAI)—An Organizational Approach to the Synthetic Modeling of Cognition based on Synthetic Biology and Embodied AI (Grant Number: 20222HHXAX). M.V. acknowledges support by Università degli Studi di Modena e Reggio Emilia (FAR2023 project of the Department of Physics, Informatics and Mathematics). Y.G. is supported by the Programma Nazionale della Ricerca (PNR) grant J95F21002830001 with title "FAIR-by-design".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: Y.G. acknowledges Area Science Park supercomputing platform ORFEO made available for conducting the research reported in this paper and the technical support of the Laboratory of Data Engineering staff.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pfeifer, R.; Bongard, J. *How the Body Shapes the Way We Think: A New View of Intelligence*; MIT Press: Cambridge, MA, USA, 2007.
2. Pfeifer, R.; Scheier, C. *Understanding Intelligence*; The MIT Press: Cambridge, MA, USA, 1999.
3. Nolfi, S.; Floreano, D. *Evolutionary Robotics*; The MIT Press: Cambridge, MA, USA, 2000.
4. Ashby, W. *Design for a Brain: The Origin of Adaptive Behaviour*, 2nd ed.; Butler & Tanner Ltd.: Leipheim, Germany, 1954.
5. Beer, R. A dynamical systems perspective on agent-environment interaction. *Artif. Intell.* **1995**, *72*, 173–215. [CrossRef]
6. Bich, L.; Moreno, A. The role of regulation in the origin and synthetic modelling of minimal cognition. *Biosystems* **2016**, *148*, 12–21. [CrossRef] [PubMed]
7. Cangelosi, A.; Asada, M. (Eds.) *Cognitive Robotics*; Intelligent Robotics and Autonomous Agents Series; The MIT Press: Cambridge, MA, USA, 2022.
8. Cariani, P. To evolve an ear. epistemological implications of gordon pask's electrochemical devices. *Syst. Res.* **1993**, *10*, 19–33. [CrossRef]
9. Di Paolo, E. Organismically-inspired robotics: Homeostatic adaptation and teleology beyond the closed sensorimotor loop. In *Dynamical Systems Approach to Embodiment and Sociality*; Murase, K., Asakura, T., Eds.; Advanced Knowledge International: Adelaide, Australia, 2003; pp. 19–42.
10. Hoffmeyer, J. *Biosemiotics: An Examination into the Signs of Life and the Life of Signs*; University of Chicago Press: Chicago, IL, USA, 2008.
11. Kull, K. On semiosis, umwelt, and semiosphere. *Semiotica* **1998**, *120*, 299–310.
12. Olsson, L.; Nehaniv, C.; Polani, D. From unknown sensors and actuators to actions grounded in sensorimotor perceptions. *Connect. Sci.* **2006**, *18*, 121–144. [CrossRef]
13. Pfeifer, R.; Lungarella, M.; Iida, F. Self-organization, embodiment, and biologically inspired robotics. *Science* **2007**, *318*, 1088–1093. [CrossRef] [PubMed]
14. Pezzulo, G.; Rigoli, F.; Friston, K. Active inference, homeostatic regulation and adaptive behavioural control. *Prog. Neurobiol.* **2015**, *134*, 17–35. [CrossRef] [PubMed]
15. Harvey, I. Homeostasis and rein control: From daisyworld to active perception. In *Artificial Life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems, Boston, MA, USA, 12 September 2004*; The MIT Press: Cambridge, MA, USA, 2004.
16. Braccini, M.; Roli, A.; Barbieri, E.; Kauffman, S. On the criticality of adaptive boolean network robots. *Entropy* **2022**, *24*, 1368. [CrossRef] [PubMed]
17. Roli, A.; Manfroni, M.; Pinciroli, C.; Birattari, M. On the design of Boolean network robots. In *Applications of Evolutionary Computation*; Di Chio, C., Cagnoni, S., Cotta, C., Ebner, M., Ekárt, A., Esparcia-Alcázar, A., Merelo, J., Neri, F., Preuss, M., Richter, H., et al., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6624, pp. 43–52.
18. Kauffman, S. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* **1969**, *22*, 437–467. [CrossRef] [PubMed]

19. Kauffman, S. *The Origins of Order: Self-Organization and Selection in Evolution*; Oxford University Press: Oxford, UK, 1993.
20. Bastolla, U.; Parisi, G. A numerical study of the critical line of Kauffman networks. *J. Theor. Biol.* **1997**, *187*, 117–133. [CrossRef] [PubMed]
21. Derrida, B.; Pomeau, Y. Random networks of automata: A simple annealed approximation. *Europhys. Lett.* **1986**, *1*, 45–49. [CrossRef]
22. Hidalgo, J.; Grilli, J.; Suweis, S.; Muñoz, M.; Banavar, J.; Maritan, A. Information-based fitness and the emergence of criticality in living systems. *Proc. Natl. Acad. Sci. USA* **2014**, *111*, 10095–10100. [CrossRef] [PubMed]
23. Mora, T.; Bialek, W. Are biological systems poised at criticality? *J. Stat. Phys.* **2011**, *144*, 268–302. [CrossRef]
24. Muñoz, M. Colloquium: Criticality and dynamical scaling in living systems. *Rev. Mod. Phys.* **2018**, *90*, 031001. [CrossRef]
25. Teuscher, C. Revisiting the edge of chaos: Again? *Biosystems* **2022**, *218*, 104693. [CrossRef] [PubMed]
26. Balleza, E.; Alvarez-Buylla, E.; Chaos, A.; Kauffman, S.; Shmulevich, I.; Aldana, M. Critical dynamics in genetic regulatory networks: Examples from four kingdoms. *PLoS ONE* **2008**, *3*, e2456. [CrossRef] [PubMed]
27. Braccini, M.; Roli, A.; Kauffman, S. A novel online adaptation mechanism in artificial systems provides phenotypic plasticity. In *Artificial Life and Evolutionary Computation*; Schneider, J.J., Weyland, M.S., Flumini, D., Füchslin, R.M., Eds.; Springer Nature: Geneva, Switzerland, 2022; pp. 121–132.
28. Krawitz, P.; Shmulevich, I. Basin entropy in boolean network ensembles. *Phys. Rev. Lett.* **2007**, *98*, 158701. [CrossRef] [PubMed]
29. Nykter, M.; Price, N.; Aldana, M.; Ramsey, S.; Kauffman, S.; Hood, L.; Yli-Harja, O.; Shmulevich, I. Gene expression dynamics in the macrophage exhibit criticality. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 1897–1900. [CrossRef]
30. Serra, R.; Villani, M.; Semeria, A. Genetic network models and statistical properties of gene expression data in knock-out experiments. *J. Theor. Biol.* **2004**, *227*, 149–157. [CrossRef] [PubMed]
31. Shmulevich, I.; Kauffman, S.; Aldana, M. Eukaryotic cells are dynamically ordered or critical but not chaotic. *Proc. Natl. Acad. Sci. USA* **2005**, *102*, 13439–13444. [CrossRef] [PubMed]
32. Torres-Sosa, C.; Huang, S.; Aldana, M. Criticality is an emergent property of genetic networks that exhibit evolvability. *PLOS Comput. Biol.* **2012**, *8*, e1002669. [CrossRef] [PubMed]
33. Villani, M.; La Rocca, L.; Kauffman, S.; Serra, R. Dynamical criticality in gene regulatory networks. *Complexity* **2018**, *14*, 5980636. [CrossRef]
34. Braccini, M.; Roli, A.; Villani, M.; Serra, R. Dynamical properties and path dependence in a gene-network model of cell differentiation. *Soft Comput.* **2021**, *25*, 6775–6787. [CrossRef]
35. Brooks, R. A robust layered control system for a mobile robot. *IEEE J. Robot. Autom.* **1986**, *RA-2*, 14–23. [CrossRef]
36. Mondada, F.; Bonani, M.; Raemy, X.; Pugh, J.; Cianci, C.; Klaptocz, A.; Magnenat, S.; Zufferey, J.-C.; Floreano, D.; Martinoli, A. The e-puck, a robot designed for education in engineering. In Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, Castelo Branco, Portugal, 7 May 2009; IPCB Instituto Politécnico de Castelo Branco: Castelo Branco, Portugal, 2009; Volume 1, pp. 59–65.
37. Bredeche, N.; Haasdijk, E.; Eiben, A. On-line, on-board evolution of robot controllers. In Proceedings of the International Conference on Artificial Evolution Evolution Artificielle, Strasbourg, France, 26–28 October 2009; Springer: Berlin/Heidelberg, Germany; pp. 110–121.
38. Feola, L.; Trianni, V. Adaptive strategies for team formation in minimalist robot swarms. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4079–4085. [CrossRef]
39. Floreano, D.; Mondada, F. Evolution of homing navigation in a real mobile robot. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **1996**, *26*, 396–407. [CrossRef] [PubMed]
40. Francesca, G.; Brambilla, M.; Brutschy, A.; Garattoni, L.; Miletitch, R.; Podevijn, G.; Reina, A.; Soleymani, T.; Salvaro, M.; Pinciroli, C.; et al. Automode-chocolate: Automatic design of control software for robot swarms. *Swarm Intell.* **2015**, *9*, 125–152. [CrossRef]
41. Howard, D.; Eiben, A.; Kennedy, D.; Mouret, J.-B.; Valencia, P.; Winkler, D. Evolving embodied intelligence from materials to machines. *Nat. Mach. Intell.* **2019**, *1*, 12–19. [CrossRef]
42. Salman, M.; Garzón Ramos, D.; Birattari, M. Automatic design of stigmergy-based behaviours for robot swarms. *Commun. Eng.* **2024**, *3*, 30. [CrossRef]
43. Parr, T.; Pezzulo, G.; Friston, K. *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*; MIT Press: Cambridge, MA, USA, 2022.
44. Baldini, P.; Roli, A.; Braccini, M. On the performance of online adaptation of robots controlled by nanowire networks. *IEEE Access* **2023**, *11*, 144408–144420. [CrossRef]
45. Cariani, P. Creating new informational primitives in minds and machines. In *Computers and Creativity*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 383–417.
46. Cariani, P. The homeostat as embodiment of adaptive control. *Int. J. Gen. Syst.* **2009**, *38*, 139–154. [CrossRef]
47. Man, K.; Damasio, A. Homeostasis and soft robotics in the design of feeling machines. *Nat. Mach. Intell.* **2019**, *1*, 446–452. [CrossRef]

sensors

MDPI

Article

Reliability of Obstacle-Crossing Parameters during Overground Walking in Young Adults

Matthias Chardon [1,2], Fabio Augusto Barbieri [2,*], Pascal Petit [1] and Nicolas Vuillerme [1,3,*]

1 AGEIS (Autonomie, Gérontologie, E-Santé, Imagerie et Société), Université Grenoble Alpes, 38000 Grenoble, France; matthias.chardon@gmail.com (M.C.); pascal.petit@univ-grenoble-alpes.fr (P.P.)
2 Human Movement Research Laboratory (MOVI-LAB), Department of Physical Education, Sao Paulo State University (UNESP), Bauru 17033-360, SP, Brazil
3 Institut Universitaire de France, 75005 Paris, France
* Correspondence: fabio.barbieri@unesp.br (F.A.B.); nicolas.vuillerme@univ-grenoble-alpes.fr (N.V.)

Abstract: We aimed to evaluate the intra-session relative and absolute reliability of obstacle-crossing parameters during overground walking in young adults, and to determine the number of trials required to ensure reliable assessment. We analysed data from 43 young male adults who were instructed to walk at a self-selected velocity on a pathway and to step over an obstacle (height = 15 cm; width = 80 cm, thickness = 2 cm) three times. Spatial–temporal gait parameters of the approaching and crossing phases (i.e., before and after the obstacle) and obstacle clearance parameters (i.e., vertical and horizontal distance between the foot and the obstacle during crossing) were computed using a three-dimensional motion analysis system. Intraclass correlation coefficients were used to compute the relative reliability, while standard error of measurement and minimal detectable change were used to assess the absolute reliability for all possible combinations between trials. Results showed that most spatial–temporal gait parameters and obstacle clearance parameters are reliable using the average of three trials. However, the mean of the second and third trials ensures the best relative and absolute reliabilities of most obstacle-crossing parameters. Further works are needed to generalize these results in more realistic conditions and in other populations.

Keywords: reliability; obstacle-crossing; gait

Citation: Chardon, M.; Barbieri, F.A.; Petit, P.; Vuillerme, N. Reliability of Obstacle-Crossing Parameters during Overground Walking in Young Adults. *Sensors* **2024**, *24*, 3387. https://doi.org/10.3390/s24113387

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 4 April 2024
Revised: 22 May 2024
Accepted: 22 May 2024
Published: 24 May 2024

1. Introduction

Reliability does represent a crucial consideration as a methodology for application in gait analysis. Indeed, if gait analysis is intended to serve as an outcome measure for diagnostic, monitoring, or therapeutic purposes, the level of reliability for each computed gait parameter has to be established.

This is certainly one of the reasons that has prompted scientists and researchers to focus on the reliability assessment of gait measurements in both healthy [1–5] and pathological [3,6–11] populations. In previous studies conducted on overground/flat conditions in healthy patients [2] and stroke individuals [6,7], speed and stride length [2,6,7], as well as cadence and gait cycle time [7], showed good reliability. In addition, knee, hip, and ankle angles showed good to excellent reliability in healthy individuals [1] and adults with spinal cord injury [6].

However, few studies have investigated the number of trials needed to ensure reliable intra-session gait assessment [2,9,12]. For instance, Soulard et al. (2021) [2] showed that three trials were enough to ensure reliable gait assessment during the 10 m walk test in both single and dual-task conditions in healthy adults [2]. Interestingly, the common feature of the above-mentioned gait reliability studies is the assessment of unobstructed level walking tasks. However, not much is known about the reliability of gait parameters under more challenging conditions, such as crossing physical obstacles while overground walking (obstructed walking task). This observation is somewhat surprising since obstacle-crossing performance during walking has been widely studied using vision/camera-based systems in various populations, including children [13,14], older healthy adults [15,16], individuals with obesity [17], and people with Parkinson's disease [18–21], multiple sclerosis [22], traumatic brain injuries [23], or stroke [24,25]. This observation is also of great concern since obstacles are likely encountered during community ambulation in an everyday environment (e.g., obstacles on sidewalks or in corridors of railway or subway stations [26]). In addition, it is widely agreed that measuring clearance parameters during the performance of the obstacle-crossing task is of significant clinical importance, assuming that lower toe clearances [27] and horizontal distances [28] could result in a high probability of foot contact and hence an increased risk of stumbles, trips, and falls.

To our knowledge, no study has yet investigated the reliability of these widely used gait and clearance parameters during obstacle-crossing in young healthy adults, whereas the reliability of gait parameters computed using vision/camera-based systems has previously been assessed during unobstructed walking in healthy adults (e.g., [29–32]). To address some of the aforementioned limitations, the purpose of the present work was to evaluate the reliability, relative and absolute, of obstacle-crossing parameters during overground walking in young adults. We further aimed to determine the number of trials required to ensure reliable obstacle-crossing measurements. Indeed, determining the number of trials required to ensure reliable measurement is a prerequisite to allow comparisons between groups at a single time point [33]. Note that this question has been the subjects of a number of studies in a variety of applications fields, including human movement, sport, and health, to name a few (e.g., [2,34–47]). In particular, intra-session reliability can favourably be used to assess the number of trials to ensure reliable measurements [2,9,38,48] to optimize gait assessments in clinical practice.

2. Materials and Methods

This study is a secondary analysis of previously published obstacle-crossing studies [22,49,50].

2.1. Participants

Data from 43 Brazilian males (age = 29.1 $\pm$ 5.91 years old, body mass = 76.3 $\pm$ 9.97 kg, body height = 1.77 $\pm$ 0.06 cm, BMI = 24.4 $\pm$ 2.72 kg/m^2, mean $\pm$ SD) were analysed. Participants had no diagnosis of muscular or neurodegenerative disease. Participants were informed about the experimental procedures and signed informed consent. The study was approved by the University Institution Review Board (authorization number: CAAE#99191318.0.0000.5398). This study and all methods were performed in accordance with the Declaration of Helsinki.

2.2. Experimental Protocol

Participants were instructed to walk barefoot at a self-selected velocity on an 8.5 m long pathway and to step over a physical obstacle (height = 15 cm; width = 80 cm, thickness = 2 cm) placed in the middle of the pathway. To ensure that participants crossed the obstacle with the right leg as the leading limb, the starting point of each trial was adjusted, and at least two strides were completed before crossing the obstacle. Participants completed three trials.

2.3. Gait Assessment

Kinematic data were recorded using ten infrared cameras (Vicon Motion System®, Oxford, UK, 200 Hz). Two markers were positioned at the top of the obstacle. Also, four markers were placed on the lateral aspect of the calcaneus and head of the second metatarsus of the right limb and the medial aspect of the calcaneus and head of the second metatarsus of the left limb. Conventional labelling of the limbs was used: the leading foot crossed the obstacle first, and the trailing foot crossed second. Signals were filtered with a low-pass (6 Hz) Butterworth filter of fifth-order (zero-lag). These pre-processed signals were used to compute the following parameters:

Spatial–temporal gait parameters (length, width, duration, velocity, and double-support time) of the approaching phase (stride before the obstacle) and of the crossing phase (step over the obstacle);

Leading and trailing foot placement prior to the obstacle (horizontal distance from the second metatarsal marker to the obstacle);

Leading and trailing vertical toe foot clearance during obstacle-crossing (vertical distance at the moment that the second metatarsal marker was above the obstacle);

Leading and trailing foot placement after the obstacle (horizontal distance from the heel marker to the obstacle).

2.4. Statistical Analysis

To examine the potential difference between trials for each obstacle-crossing parameter, a repeated measures analysis of variance (RM-ANOVA) was first carried out. In cases where a difference was observed (type I error rate set at $\alpha = 0.05$), a pairwise comparison post hoc test was performed to compare differences between pairs of trials, using the Benjamini–Hochberg approach to account for multiple testing.

To assess relative reliability across the three trials, the intraclass correlation coefficient (ICC) based on a two-way random effects model (absolute agreement, ICC(2.1)) was calculated. ICC(2.1) values inferior to 0 were considered 'poor', between 0.01 and 0.20 as 'slight', between 0.21–0.40 as 'fair', between 0.41–0.60 as 'moderate', between 0.61–0.80 as 'substantial', and 0.80–1.00 as 'almost perfect' relative reliability [51].

To assess absolute reliability across the three trials, the standard error of measurement (SEM) and the minimum detectable change (MDC) were computed. SEM, with the same unit as obstacle-crossing parameters, corresponds to the absolute measure of the variability of the errors of measurements and informs on the precision of obstacle-crossing parameters of individual participants [52]. SEM was calculated using the following formula [53]: $SEM = SD\sqrt{(1\text{-}ICC(2.1))}$, where SD is the standard deviation of the obstacle-crossing parameters from all participants and ICC(2.1) is the relative reliability. MDC is the minimum value for which a difference can be considered "real". MDC was calculated using the following formula [54]: $MDC = SEM \times 1.96 \times \sqrt{2}$. The SEM% and MDC% were also expressed as a percentage of the mean for each obstacle-crossing parameter. Lower values for SEM% and MDC% indicate higher absolute reliability. Precisely, SEM% values were considered 'low' (SEM% $\leq$ 10%) or 'high' (SEM% > 10%) [55], while MDC% values were considered 'low' (MDC% $\leq$ 20%), 'acceptable' (20% < MDC% < 40%), or 'high' (MDC% $\geq$ 40%) [56].

To quantify the agreement between pairs of trials for each of the 16 obstacle-crossing parameters, Lin's concordance correlation coefficient (CCC) and associated 95% confidence intervals [57,58], as well as the Shieh exact test for agreement [59], were used. The CCC indicates how close the measurement pairs fall to the 45-degree line (perfect agreement). Hence, the CCC is a measure of the reproducibility of measurements. The CCC ranges from -1 to 1, with perfect agreement at 1 and perfect discordance at -1. Agreement was considered 'poor' for CCC values < 0.40, 'moderate' for CCC values ranging from 0.40 to 0.70, and 'good' for CCC values > 0.70 [60].

Limits of agreement (LOA) and Bland–Altman plots of the differences between trials and their arithmetic mean were used to assess the magnitude of disagreement between trials for each obstacle-crossing parameter. LOA allows the quantification of bias and the determination of a range of agreement, within which 95% of the differences between one measurement and the other are included. The bias is significant when the line of equality is not within the 95% CI of the mean difference. The smallest worthwhile change (SWC) was used to determine the maximum allowed difference between trials presented in Bland–Altman plots [61]. The SWC provides a method to evaluate a real change in performance between trials. If the SWC is low and included in the LOA or Shieh 95% CI, it indicates a low variation between trials. Two trials are considered in agreement if the LOA or Shieh exact test does not exceed the maximum allowed difference between trials (SWC).

All statistical analyses were performed using R software 4.3.1® (R Core Team, Vienna, Austria) for Windows 10©. The list of all R packages used is provided in the Supplementary Materials.

3. Results

3.1. Difference between Trials

Most obstacle-crossing parameters exhibited no significant differences between trials. The only three exceptions were observed for the stride width and the double support time (p = 0.04, eta-squared = 0.01) during the approaching phase, and for the trailing foot horizontal distance after the obstacle (p = 0.02, eta-squared = 0.02). For these three parameters, pairwise t-tests showed that trial 1 (T1) differed from trials 2 and 3. Stride width was lower in T1 compared to T2 and T3 (p = 0.02 for T1–T2, p = 0.04 for T1–T3), double support time was greater for T1 compared to T2 and T3 (p = 0.05 for T1–T2, p = 0.03 for T1–T3), while for the trailing foot horizontal distance after obstacle, T3 was higher compared to T1 and T2 (p = 0.03).

Table 1 provides the arithmetic mean and standard deviation of obstacle-crossing parameters for each trial and the mean of the three trials.

Table 1. Arithmetic mean and standard deviation of obstacle-crossing parameters for each trial and for their mean.

Condition	Parameter	T1	T2	T3	T123
		Mean (SD)	**Mean (SD)**	**Mean (SD)**	**Mean (SD)**
Approaching phase	Stride double support time (s)	0.36 (0.05)	0.35 (0.06)	0.35 (0.05)	0.36 (0.05)
	Stride duration (s)	1.10 (0.13)	1.09 (0.11)	1.09 (0.10)	1.09 (0.11)
	Stride length (cm)	131 (11.4)	131 (11.5)	132 (11.2)	131 (11.3)
	Stride velocity (cm/s)	120 (17.2)	122 (15.5)	122 (15.4)	121 (15.9)
	Stride width (cm)	10.6 (2.49)	11.3 (2.74)	11.2 (2.86)	11.0 (2.70)
Crossing phase	Step double support time (s)	0.22 (0.04)	0.20 (0.05)	0.21 (0.04)	0.21 (0.04)
	Step duration (s)	0.64 (0.08)	0.62 (0.07)	0.63 (0.07)	0.63 (0.07)
	Step length (cm)	72.7 (9.49)	73.4 (9.78)	73.6 (9.33)	73.2 (9.47)
	Step velocity (cm/s)	115 (20.8)	119 (21.5)	118 (19.0)	118 (20.4)
	Step width (cm)	10.2 (3.40)	10.4 (3.80)	10.1 (3.79)	10.2 (3.64)
Obstacle clearance	Leading foot horizontal distance prior to obstacle (cm)	108 (13.0)	108 (10.4)	108 (10.1)	108 (11.1)
	Trailing foot horizontal distance prior to obstacle (cm)	43.0 (10.3)	44.1 (8.77)	42.9 (10.1)	43.3 (9.71)
	Leading foot horizontal distance after obstacle (cm)	25.7 (6.61)	25.0 (6.63)	26.8 (7.75)	25.8 (7.00)
	Trailing foot horizontal distance after obstacle (cm)	91.6 (11.5)	91.8 (10.4)	95.0 (12.1)	92.8 (11.4)
	Leading foot toe clearance (cm)	9.93 (4.21)	9.99 (4.37)	10.8 (5.19)	10.2 (4.59)
	Trailing foot toe clearance (cm)	31.4 (8.68)	32.2 (8.01)	32.0 (6.53)	31.9 (7.74)

Note: SD: arithmetic standard deviation, T1: trial 1, T2: trial 2, T3: trial 3, T123: arithmetic mean of the three trials.

3.2. Absolute and Relative Reliabilities

Figure 1 and Table 2 present the ICC(2.1) values for each obstacle-crossing parameter for all comparisons of trials. Table 2 also provides the SEM, SEM%, MDC, and MDC%.

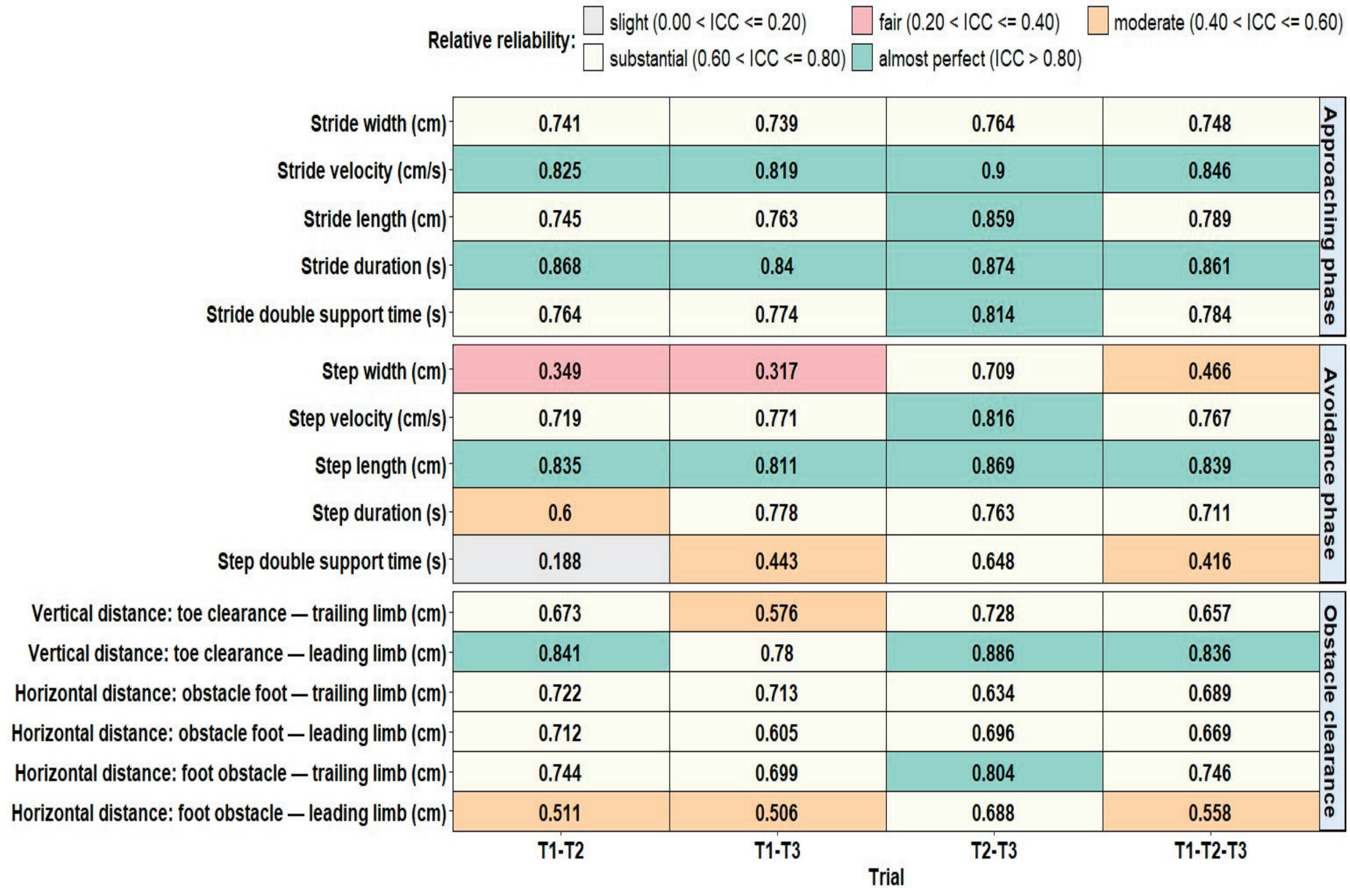


Figure 1. Mean ICC(2.1) for the means of trial 1–2, 1–3, 2–3 and 1–2–3 calculated for each obstacle-crossing parameter.

Table 2. Relative and absolute reliability for each obstacle-crossing parameter between the mean of the first and second trials (T1–T2), the first and third trials (T1–T3), the second and third trials (T2–T3), and the means of the three consecutive trials (T1–T2–T3).

Condition	Parameter	T1–T2					T1–T3					T2–T3					T1–T2–T3				
		ICC(2.1) [95%CI]	MDC	MDC%	SEM	SEM%	ICC [95%CI]	MDC	MDC%	SEM	SEM%	ICC(2.1) [95%CI]	MDC	MDC%	SEM	SEM%	ICC(2.1) [95%CI]	MDC	MDC%	SEM	SEM%
Approaching phase	Stride length (cm)	0.75 [0.58; 0.85]	15.9	12.2	5.75	4.40	0.76 [0.60; 0.86]	15.2	11.6	5.48	4.18	0.86 [0.76; 0.92]	11.7	8.94	4.23	3.23	0.79 [0.68; 0.87]	14.4	11.0	5.19	3.96
	Stride width (cm)	0.74 [0.55; 0.85]	3.71	34.0	1.34	12.3	0.74 [0.56; 0.85]	3.81	35.0	1.37	12.6	0.76 [0.61; 0.87]	3.75	33.4	1.35	12.1	0.75 [0.62; 0.84]	3.76	34.2	1.35	12.3
	Stride double support time (s)	0.76 [0.60; 0.87]	0.07	20.4	0.03	7.36	0.77 [0.61; 0.87]	0.07	18.8	0.02	6.78	0.81 [0.68; 0.89]	0.06	18.3	0.02	6.60	0.78 [0.67; 0.87]	0.07	19.2	0.02	6.92
	Stride duration (s)	0.87 [0.77; 0.93]	0.12	10.9	0.04	3.95	0.84 [0.72; 0.91]	0.13	11.5	0.05	4.15	0.87 [0.78; 0.93]	0.11	9.73	0.04	3.51	0.86 [0.78; 0.92]	0.12	10.7	0.04	3.88
	Stride velocity (cm/s)	0.83 [0.70; 0.90]	18.9	15.6	6.82	5.64	0.82 [0.69; 0.90]	19.2	15.8	6.92	5.71	0.90 [0.82; 0.95]	13.4	11.0	4.85	3.98	0.85 [0.76; 0.91]	17.3	14.3	6.26	5.16
Crossing phase	Step length (cm)	0.84 [0.72; 0.91]	10.8	14.8	3.89	5.33	0.81 [0.68; 0.89]	11.3	15.4	4.07	5.57	0.87 [0.77; 0.93]	9.52	13.0	3.44	4.67	0.84 [0.75; 0.90]	10.5	14.4	3.80	5.19
	Step width (cm)	0.35 [0.06; 0.59]	8.02	77.9	2.89	28.1	0.32 [0.02; 0.56]	8.20	80.8	2.96	29.2	0.71 [0.52; 0.83]	5.64	54.8	2.03	19.8	0.47 [0.28; 0.64]	7.37	72.0	2.66	26.0
	Step double support time (s)	0.19 [−0.10; 0.45]	0.11	53.5	0.04	19.3	0.44 [0.18; 0.65]	0.09	40.9	0.03	14.8	0.65 [0.44; 0.79]	0.07	34.4	0.03	12.4	0.42 [0.23; 0.60]	0.09	43.9	0.03	15.8
	Step duration (s)	0.60 [0.37; 0.76]	0.13	20.3	0.05	7.34	0.78 [0.63; 0.87]	0.09	14.9	0.03	5.38	0.76 [0.61; 0.86]	0.09	14.8	0.03	5.34	0.71 [0.58; 0.82]	0.11	16.9	0.04	6.09
	Step velocity (cm/s)	0.72 [0.54; 0.84]	31.0	26.5	11.2	9.54	0.77 [0.62; 0.87]	26.3	22.5	9.49	8.14	0.82 [0.69; 0.90]	24.0	20.2	8.65	7.29	0.77 [0.65; 0.86]	27.3	23.2	9.83	8.36
Obstacle clearance	Trailing foot horizontal distance prior to obstacle (cm)	0.74 [0.58; 0.85]	13.4	30.8	4.83	11.1	0.70 [0.51; 0.82]	15.5	36.0	5.58	13.0	0.80 [0.67; 0.89]	11.6	26.6	4.18	9.60	0.75 [0.62; 0.84]	13.6	31.3	4.89	11.3
	Trailing foot horizontal distance after obstacle (cm)	0.72 [0.54; 0.84]	15.9	17.4	5.74	6.26	0.71 [0.51; 0.84]	17.6	18.9	6.35	6.81	0.63 [0.41; 0.79]	18.9	20.3	6.83	7.32	0.69 [0.54; 0.80]	17.6	18.9	6.33	6.82
	Trailing foot toe clearance (cm)	0.67 [0.47; 0.81]	13.2	41.4	4.76	15.0	0.58 [0.34; 0.75]	13.8	43.5	4.98	15.7	0.73 [0.55; 0.84]	10.5	32.7	3.79	11.8	0.66 [0.51; 0.78]	12.6	39.4	4.53	14.2
	Leading foot horizontal distance prior to obstacle (cm)	0.51 [0.25; 0.70]	22.6	21.0	8.17	7.57	0.51 [0.25; 0.70]	22.5	20.9	8.12	7.55	0.69 [0.49; 0.82]	15.8	14.6	5.70	5.28	0.56 [0.39; 0.71]	20.5	19.1	7.41	6.88
	Leading foot horizontal distance after obstacle (cm)	0.71 [0.53; 0.83]	9.80	38.6	3.54	13.9	0.61 [0.38; 0.76]	12.5	47.6	4.51	17.2	0.70 [0.50; 0.82]	11.0	42.6	3.98	15.4	0.67 [0.52; 0.79]	11.2	43.2	4.03	15.6
	Leading foot toe clearance (cm)	0.84 [0.73; 0.91]	4.72	47.4	1.70	17.1	0.78 [0.63; 0.88]	6.14	59.3	2.21	21.4	0.89 [0.79; 0.94]	4.47	43.0	1.61	15.5	0.84 [0.75; 0.90]	5.16	50.4	1.86	18.2

Note: ICC: intraclass correlation coefficient, MDC: minimal detectable change, SEM: standard error of measurement, T1: trial 1, T2: trial 2, T3: trial 3. Mean relative reliability legend: slight (gray: 0.00 < ICC(2.1) < 0.20), fair (red: 0.21 < ICC(2.1) < 0.40), moderate (orange: 0.41 < ICC(2.1) < 0.60), substantial (yellow: 0.61 < ICC(2.1) < 0.80), and almost perfect (green: 0.81 < ICC(2.1) < 1.00). MDC% legend: low (yellow: MDC % ≤ 20%), acceptable (orange: 20% < MDC% < 40%), and high (red: MDC% ≥ 40%). SEM% legend: low (green: SEM % ≤ 10%), and high (SEM% > 10%).

Overall, when pooling the values of the three trials, the relative reliability ranged from moderate to practically perfect, whereas the absolute reliability was good for nine out of the sixteen parameters. Furthermore, the highest relative and absolute reliabilities were found when pooling values from T2–T3, followed by T1–T2–T3, T1–T3, and T1–T2.

3.2.1. Spatial–Temporal Gait Parameters during the Approaching Phase

Relative reliability: Regardless of the comparison of trials considered, spatial–temporal gait parameters during the approaching phase showed either substantial or almost perfect reliability. Indeed, ICC(2.1) ranged from 0.61 to 0.80 for stride width for all trial comparisons, for stride length, and double support time for T1–T2, T1–T3, and T1–T2–T3, while ICC(2.1) ranged from 0.81 to 1.00 for stride duration and stride velocity for all trial comparisons, and for stride length and double support time for T2–T3.

Absolute reliability: Regardless of the comparison of trials considered, SEM% and MDC% values were low for most of the spatial–temporal gait parameters during the approaching phase (SEM% < 10 and MDC% < 20%), with the exception of the stride width (SEM% > 10% and 20% < MDC% < 40%, for all trial comparisons), and MDC% of the double support time (MDC% = 20.4%, for T1–T2).

Number of trials to ensure reliable measurements: The comparisons of means of ICC(2.1) between trials showed that, for all spatial–temporal gait parameters during the approaching phase, slightly higher ICC(2.1) and lower SEM% and MDC% were obtained when pooling the second and third trials, compared to the first and second trials, the first and third trials, or the three trials.

3.2.2. Spatial–Temporal Gait Parameters during the Crossing Phase

Relative reliability: Regardless of the comparison of trials considered, step length, step duration, and step velocity showed either substantial or almost perfect relative reliability. Indeed, ICC(2.1) ranged from 0.61 to 0.80, for step duration for all trial comparisons and for step velocity for T1–T2, T1–T3, and T1–T2–T3, while ICC(2.1) ranged from 0.81 to 1.00, for step length for all trial comparisons and for step velocity for T2–T3. Furthermore, step width and double support time showed slight relative reliability (ICC(2.1) = 0.19 for double support time for T1–T2), fair relative reliability (ICC(2.1) ranged from 0.21 to 0.40, for step width for T1–T2 and T1–T3), moderate relative reliability (ICC(2.1) ranged from 0.41 to 0.60 for step width for T1–T2–T3, and for double support time for T1–T3 and T1–T2–T3), or substantial relative reliability (ICC(2.1) ranged from 0.61 to 0.80 for step width and double support time for T2–T3 only).

Absolute reliability: Regardless of the comparison of trials considered, SEM% and MDC% were low or acceptable (SEM% < 10% and MDC% < 40%) for step length, duration, and velocity. Step width and double support time showed high SEM% and MDC% values (SEM% > 10% and MDC% > 40%), except for double support time when pooling the second and third trials with an acceptable MDC% (MDC% = 34.4%).

Number of trials to ensure reliable measurements: The comparisons of means of ICC(2.1) between trials showed that, for almost all spatial–temporal gait parameters during the crossing phase, slightly higher ICC(2.1) (except for step duration, the best relative reliability was found for T1–T3, ICC(2.1) = 0.78) and lower SEM% (all parameters) and MDC% (all parameters) were obtained when pooling the second and third trials, compared to the first and second trials, the first and third trials, or the three trials.

3.2.3. Obstacle Clearance Parameters

Relative reliability: Regardless of the comparison of trials considered, obstacle clearance parameters showed either moderate, substantial, or almost perfect mean relative reliability. The highest ICC(2.1) values were found for leading toe clearance with substantial to almost perfect reliability ($0.78 \leq$ ICC(2.1) ≤ 0.89 for all trial comparisons), followed by trailing foot horizontal distance prior to the obstacle, and leading and trailing foot horizontal distance after the obstacle with substantial reliability ($0.61 \leq$ ICC(2.1) ≤ 0.80 for all trial comparisons). The lowest ICC values were found for the trailing foot toe clearance (moderate reliability, ICC(2.1) = 0.58, for T1–T3) and for the leading foot horizontal distance prior to an obstacle (moderate reliability, $0.51 \leq$ ICC(2.1) ≤ 0.56, for T1–T2, T1–T3, and T1–T2–T3).

Absolute reliability: Regardless of the comparison of trials considered, SEM% and MDC% were low or acceptable for leading foot horizontal distance prior to the obstacle and trailing foot horizontal distance after the obstacle (SEM% < 10% and MDC% < 40%). Regardless of the comparison of trials considered, high SEM% values were found for leading and trailing foot toe clearance, leading foot horizontal distance after the obstacle, and trailing foot horizontal distance prior to the obstacle (for T1–T2, T1–T3, and T1–T2–T3). High MDC% (MDC% > 40%) values were observed for the leading (all trial comparisons) and trailing (T1–T2, T1–T3) limb toe clearances, as well as for the horizontal distance after the obstacle for the leading limb (T1–T3, T2–T3, T1–T2–T3).

Number of trials to ensure reliable measurements. The comparisons of means of ICC(2.1) between trials showed that, for almost all clearance parameters, slightly higher ICC(2.1) and lower SEM% and MDC% were obtained when pooling the second and third trials, compared to the first and second trials, the first and the third trials, or the three trials.

3.3. Agreement between Trials

When considering the maximum allowed difference between trials (SWC), regardless of the obstacle-crossing parameter, all pairs of trials were considered in agreement because the SWC was always within the LOA and Shieh ranges.

3.3.1. Spatial–Temporal Gait Parameters during the Approaching Phase

Good agreement (CCC > 0.70) between each pair of trials was found for all spatial–temporal gait parameters during the approaching phase.

The best results were found for T2–T3 for all parameters, followed by T1–T2 and T1–T3 for stride width, stride duration, and stride velocity, or followed by T1–T3 and T1–T2 for stride length and double support time.

3.3.2. Spatial–Temporal Gait Parameters during the Crossing Phase

Moderate (0.40 < CCC < 0.70) and good (CCC > 0.70) agreements between each pair of trials were found for all spatial–temporal gait parameters during the crossing phase, with the exception of the step width for T1–T2 (CCC = 0.34) and T1–T3 (CCC = 0.31), and of the double support time for T1–T2 (CCC = 0.19) with poor agreement between trials.

The best results were found when pooling values from T2–T3 for step length, step width, step velocity, and double support time (except for step duration, agreements between trials were better for T1–T3, CCC = 0.77 vs. CCC = 0.76 for T2–T3), followed by T1–T2 and T1–T3 for step length and step width, or followed by T1–T3 and T1–T2 for step velocity and double support time.

3.3.3. Obstacle Clearance Parameters

Moderate (0.40 < CCC < 0.70) and good (CCC > 0.70) agreements between each pair of trials were found for all obstacle clearance parameters.

The best results were found when pooling values from T2–T3 for leading and trailing foot toe clearance and foot horizontal distance prior to the obstacle. However, for leading and trailing foot horizontal distance after obstacle, agreements between trials were better for T1–T2 (CCC = 0.71 vs. CCC = 0.69 for T2–T3 for leading horizontal distance after obstacle, and CCC = 0.72 vs. CCC = 0.63 for T2–T3 for trailing foot distance after obstacle), followed by T1–T2 and T1–T3.

Table 3 presents the CCC, LOA, Shieh exact test agreement, and SWC. Figure 2 presents the Bland–Altman of each gait and obstacle-crossing parameter.

Table 3. Agreement between the mean of the first and second trials (T1–T2), the first and third trials (T1–T3), and the second and third trials (T2–T3), for each obstacle-crossing parameter.

Condition	Parameter	T1–T2				T1–T3				T2–T3			
		CCC [95%CI]	LOA [95%CI]	Shieh Test [95%CI]	SWC	CCC [95%CI]	LOA [95%CI]	Shieh Test [95%CI]	SWC	CCC [95%CI]	LOA [95%CI]	Shieh Test [95%CI]	SWC
Approaching phase	Stride length (cm)	0.74 [0.57; 0.85]	[−16.1; 16.3]	[−18.7; 18.9]	2.14	0.76 [0.60; 0.86]	[−16.3; 14.3]	[−18.8; 16.8]	2.12	0.86 [0.75; 0.92]	[−12.8; 10.6]	[−14.7; 12.5]	2.19
	Stride width (cm)	0.74 [0.57; 0.84]	[−4.23; 2.85]	[−4.80; 3.42]	0.49	0.73 [0.57; 0.84]	[−4.30; 3.09]	[−4.90; 3.68]	0.50	0.76 [0.60; 0.86]	[−3.73; 3.89]	[−4.34; 4.50]	0.53
	Stride double support time (s)	0.76 [0.60; 0.86]	[−0.06; 0.08]	[−0.07; 0.09]	0.01	0.77 [0.62; 0.87]	[−0.05; 0.08]	[−0.06; 0.09]	0.01	0.81 [0.68; 0.89]	[−0.07; 0.07]	[−0.08; 0.08]	0.01
	Stride duration (s)	0.87 [0.77; 0.92]	[−0.10; 0.13]	[−0.12; 0.15]	0.02	0.84 [0.73; 0.90]	[−0.11; 0.14]	[−0.13; 0.16]	0.02	0.87 [0.78; 0.93]	[−0.11; 0.11]	[−0.13; 0.12]	0.02
	Stride velocity (cm/s)	0.82 [0.70; 0.90]	[−20.5; 17.5]	[−23.6; 20.5]	3.13	0.82 [0.69; 0.89]	[−21.2; 16.9]	[−24.3; 19.9]	3.11	0.90 [0.82; 0.94]	[−14.3; 13.0]	[−16.4; 15.2]	3.01
Crossing phase	Step length (cm)	0.83 [0.71; 0.91]	[−11.6; 10.1]	[−13.4; 11.9]	1.85	0.81 [0.67; 0.89]	[−12.2; 10.4]	[−14.1; 12.2]	1.79	0.87 [0.77; 0.93]	[−9.83; 9.53]	[−11.4; 11.1]	1.85
	Step width (cm)	0.34 [0.05; 0.58]	[−8.39; 7.85]	[−9.70; 9.15]	0.59	0.31 [0.01; 0.55]	[−8.29; 8.34]	[−9.63; 9.68]	0.58	0.70 [0.52; 0.83]	[−5.40; 6.00]	[−6.32; 6.92]	0.70
	Step double support time (s)	0.19 [−0.10; 0.44]	[−0.09; 0.13]	[−0.11; 0.14]	0.01	0.44 [0.17; 0.64]	[−0.07; 0.10]	[−0.09; 0.11]	0.01	0.64 [0.43; 0.79]	[−0.08; 0.06]	[−0.09; 0.08]	0.01
	Step duration (s)	0.59 [0.37; 0.75]	[−0.11; 0.14]	[−0.13; 0.16]	0.01	0.77 [0.62; 0.87]	[−0.09; 0.10]	[−0.10; 0.12]	0.01	0.76 [0.60; 0.86]	[−0.10; 0.09]	[−0.12; 0.10]	0.01
	Step velocity (cm/s)	0.71 [0.53; 0.83]	[−34.6; 26.7]	[−39.5; 31.7]	3.93	0.77 [0.61; 0.87]	[−28.9; 23.7]	[−33.1; 27.9]	3.75	0.81 [0.68; 0.89]	[−22.9; 25.6]	[−26.8; 29.5]	3.86
Obstacle clearance	Trailing foot horizontal distance prior to obstacle (cm)	0.74 [0.57; 0.85]	[−14.5; 12.3]	[−16.7; 14.5]	1.79	0.69 [0.50; 0.82]	[−15.6; 15.8]	[−18.2; 18.4]	1.88	0.80 [0.67; 0.88]	[−10.3; 12.7]	[−12.2; 14.6]	1.80
	Trailing foot horizontal distance after obstacle (cm)	0.72 [0.54; 0.83]	[−16.3; 16.1]	[−18.9; 18.7]	2.03	0.71 [0.53; 0.83]	[−20.1; 13.4]	[−22.8; 16.1]	2.20	0.63 [0.42; 0.77]	[−21.5; 15.1]	[−24.5; 18.0]	2.05
	Trailing foot toe clearance (cm)	0.67 [0.46; 0.80]	[−14.1; 12.4]	[−16.3; 14.6]	1.53	0.57 [0.34; 0.73]	[−14.6; 13.3]	[−16.8; 15.6]	1.36	0.72 [0.55; 0.84]	[−10.4; 10.9]	[−12.2; 12.6]	1.36
	Leading foot horizontal distance prior to obstacle (cm)	0.50 [0.25; 0.69]	[−23.4; 22.5]	[−27.1; 26.2]	2.04	0.50 [0.25; 0.69]	[−22.7; 23.0]	[−26.4; 26.7]	2.01	0.68 [0.48; 0.81]	[−15.4; 16.6]	[−18.0; 19.2]	1.88
	Leading foot horizontal distance after obstacle (cm)	0.71 [0.52; 0.83]	[−9.14; 10.6]	[−10.7; 12.2]	1.22	0.60 [0.38; 0.76]	[−13.6; 11.5]	[−15.6; 13.5]	1.29	0.69 [0.51; 0.82]	[−12.5; 9.02]	[−14.2; 10.7]	1.33
	Leading foot toe clearance (cm)	0.84 [0.72; 0.91]	[−4.86; 4.73]	[−5.63; 5.51]	0.82	0.78 [0.64; 0.87]	[−6.88; 5.18]	[−7.85; 6.15]	0.89	0.88 [0.81; 0.93]	[−5.07; 3.50]	[−5.76; 4.19]	0.93

Note: CCC: Lin's concordance correlation coefficient, LOA: limits of agreement, SWC: smallest worthwhile change, T1: trial 1, T2: trial 2, T3: trial 3. CCC legend: Poor (red: CCC $\leq$ 0.40), moderate (yellow: 0.40 < CCC < 0.70), and good (green: CCC $\geq$ 0.70) agreement.

(A)

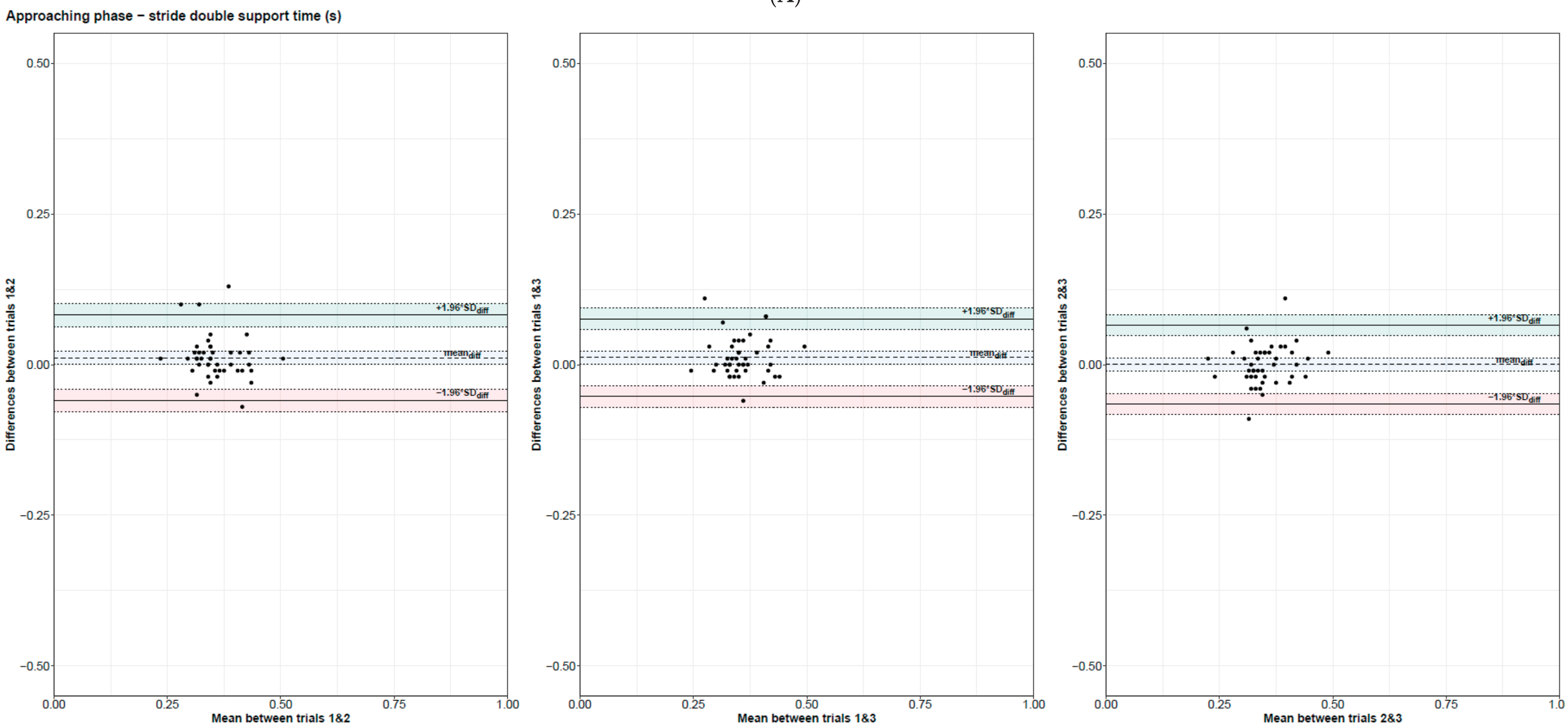


Figure 2. *Cont.*

(B)

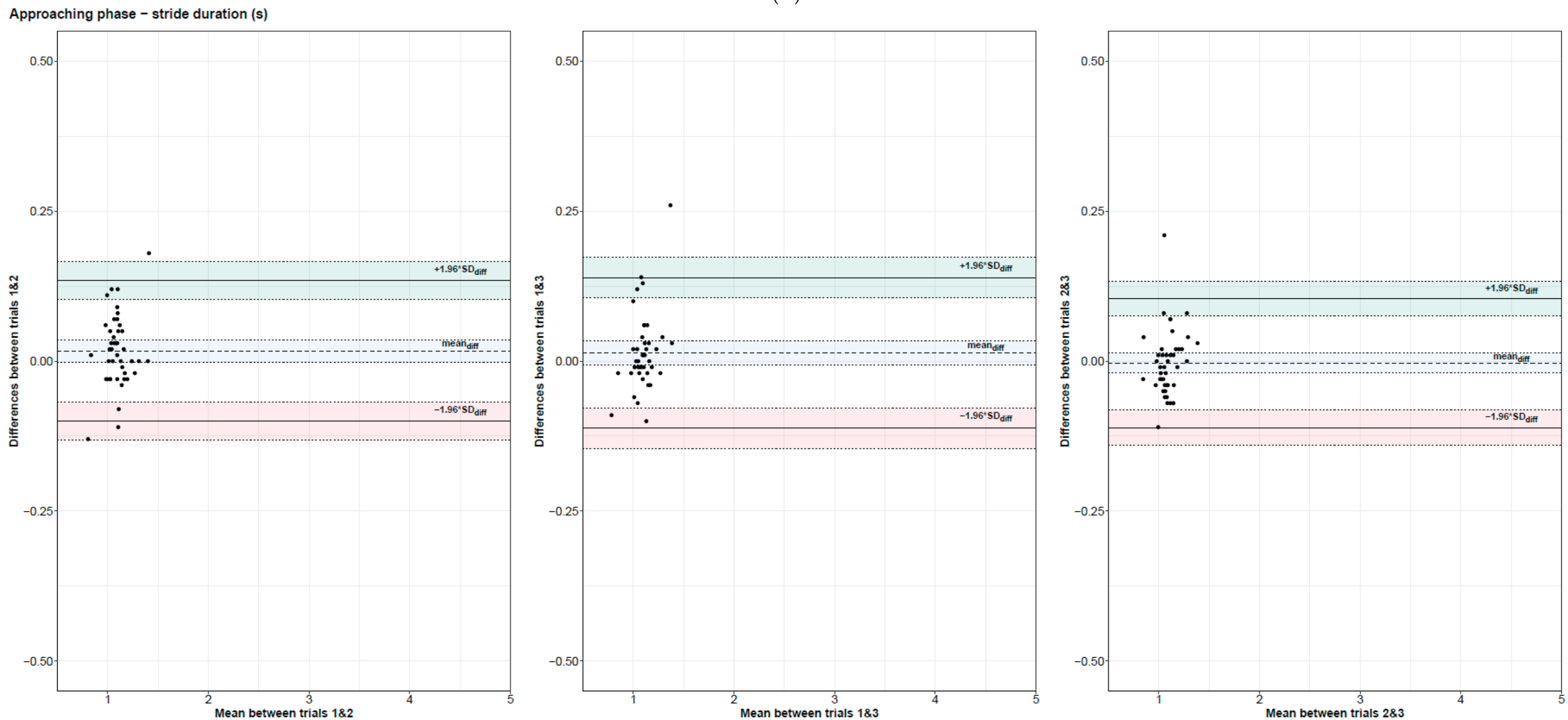


Figure 2. *Cont.*

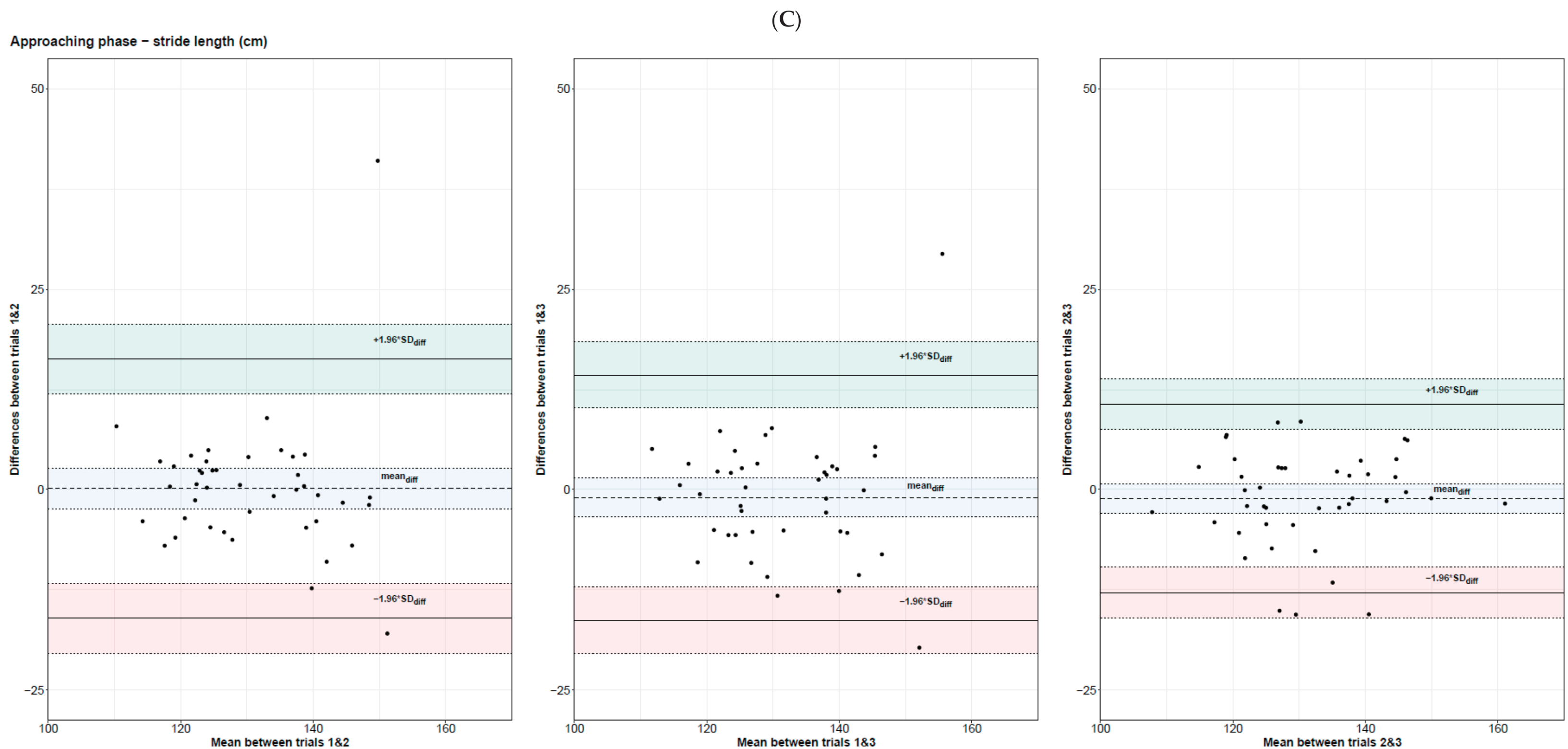


Figure 2. *Cont.*

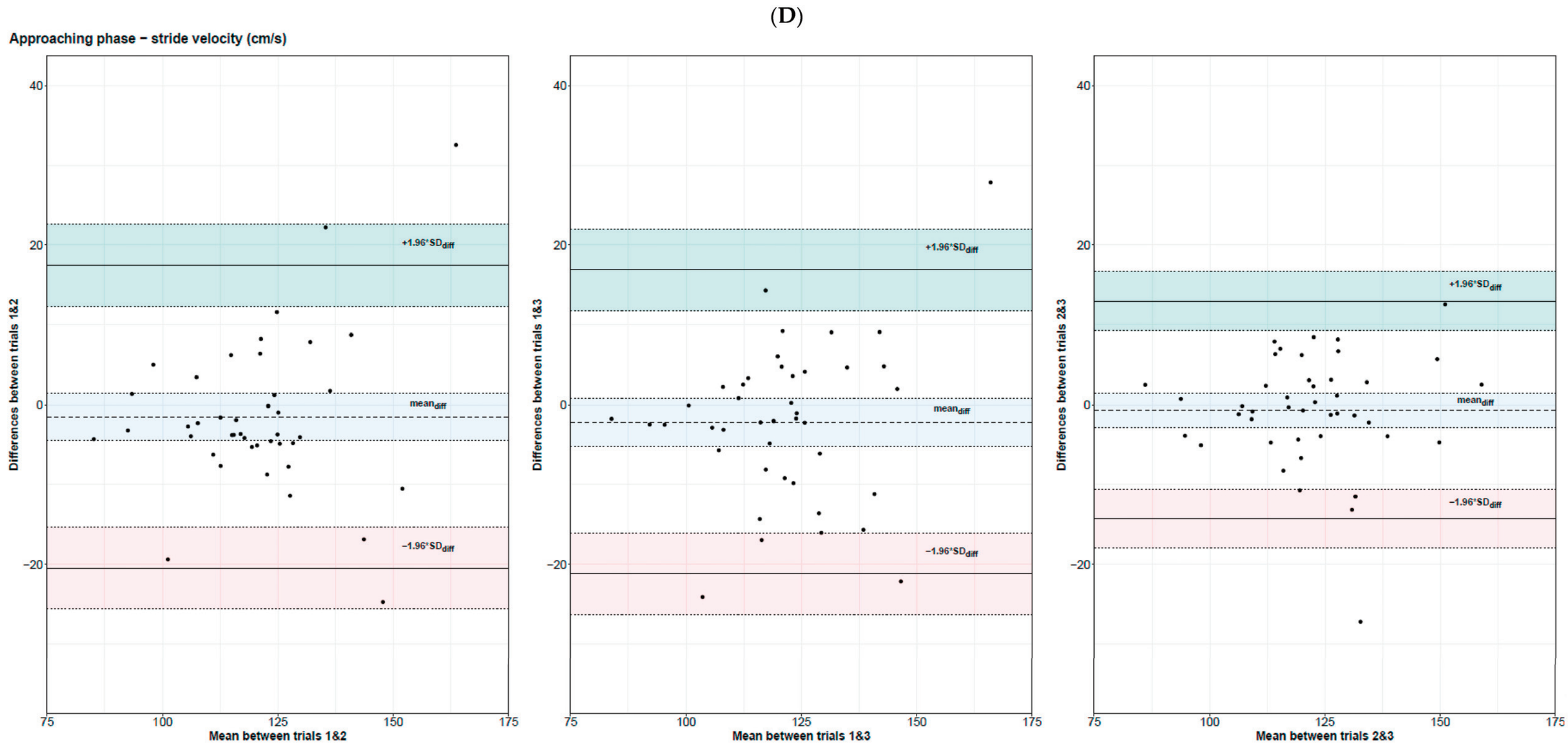


Figure 2. *Cont.*

(E)

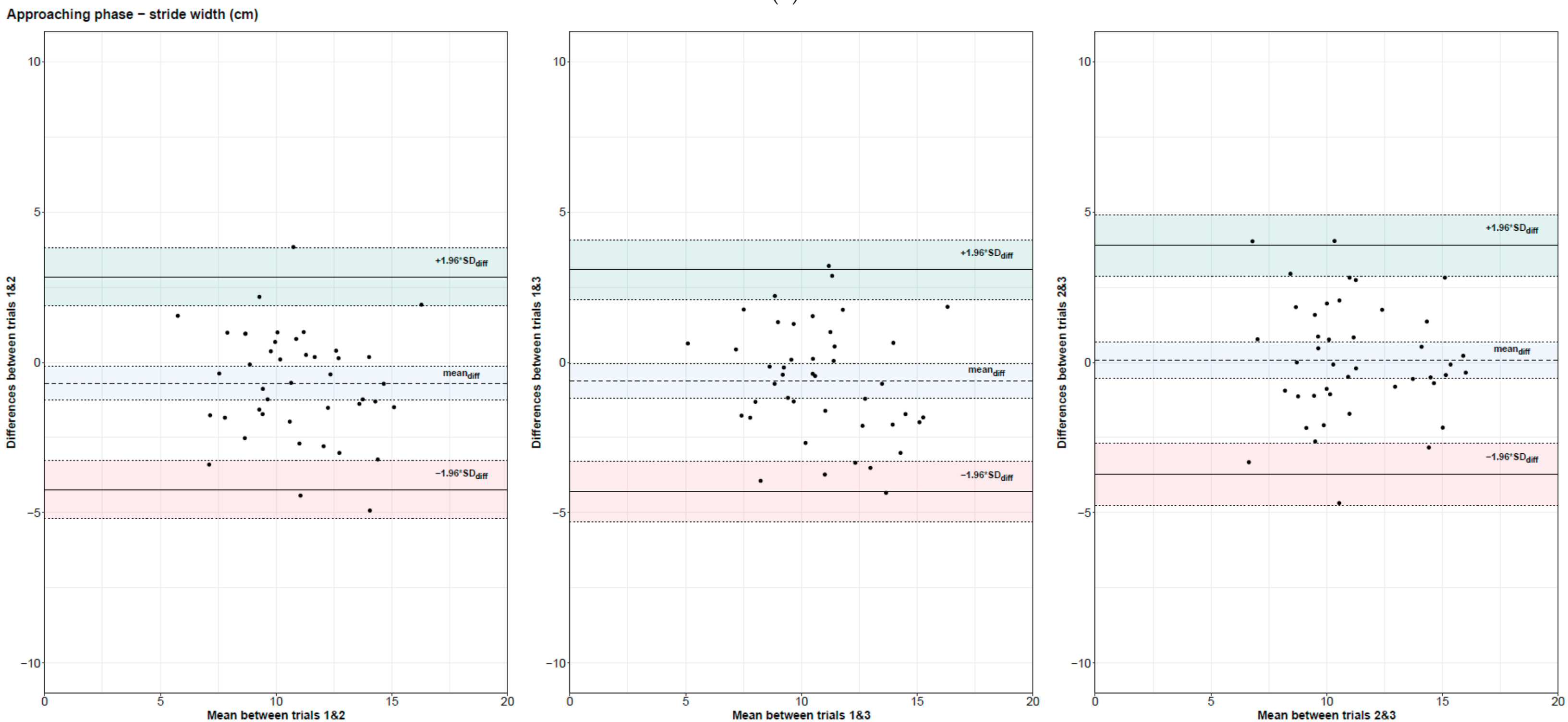


Figure 2. *Cont.*

(F)

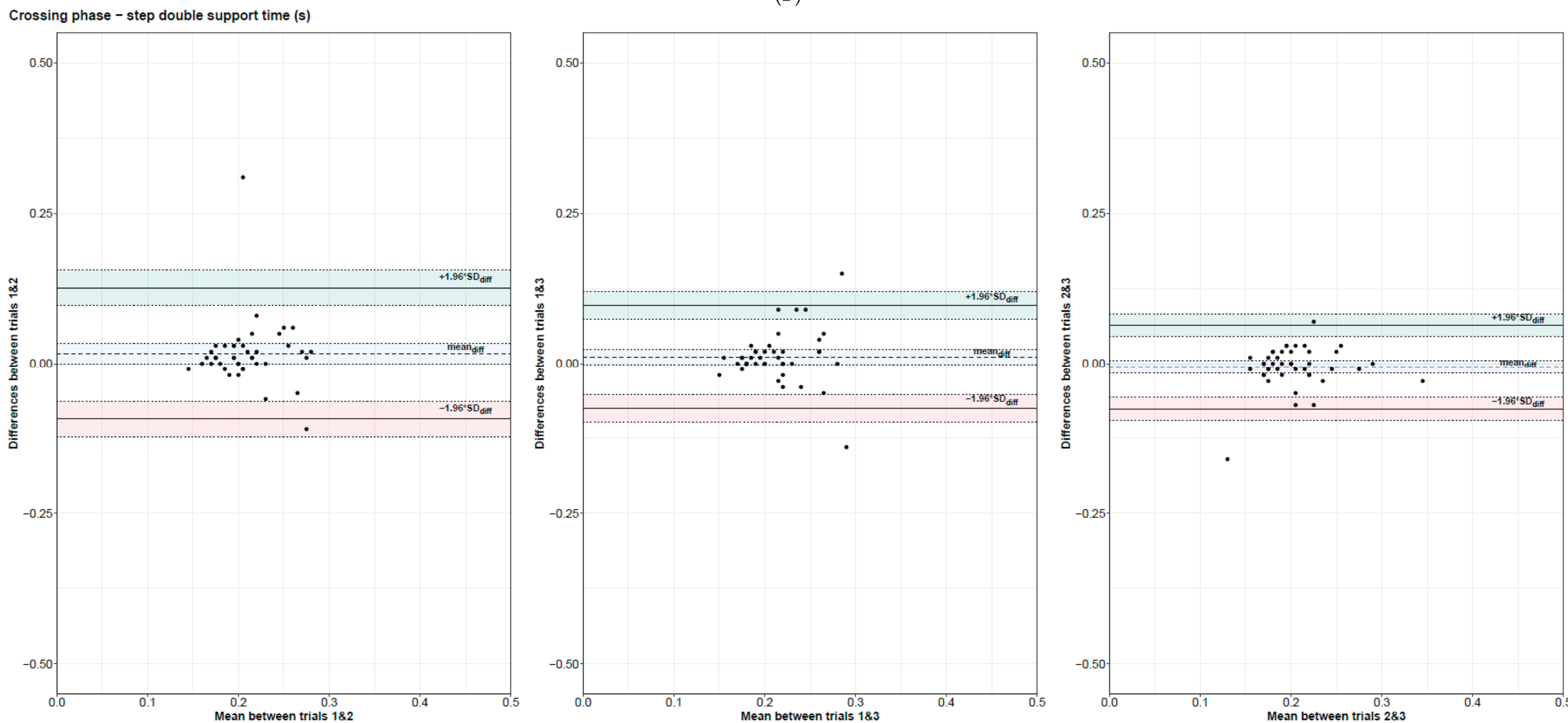


Figure 2. *Cont.*

(G)

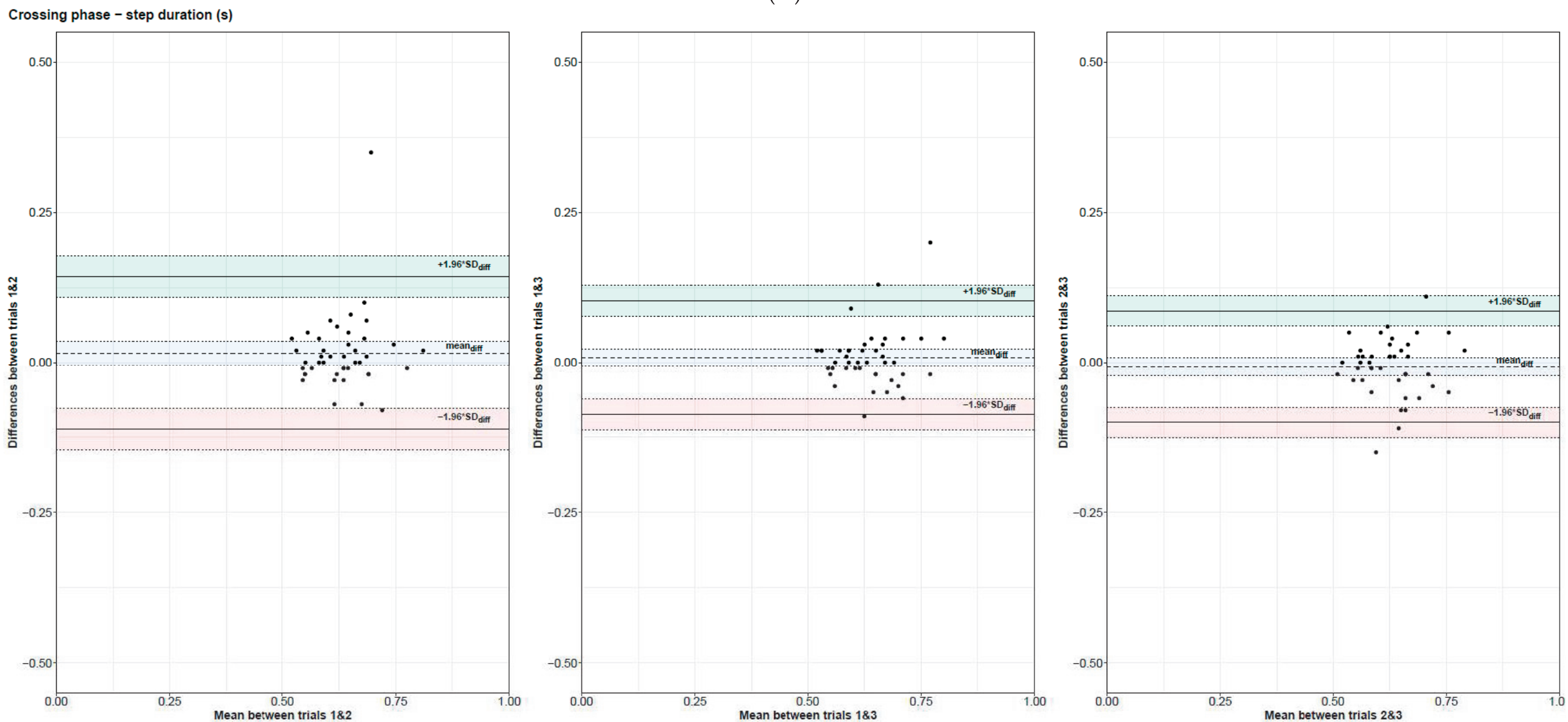


Figure 2. *Cont.*

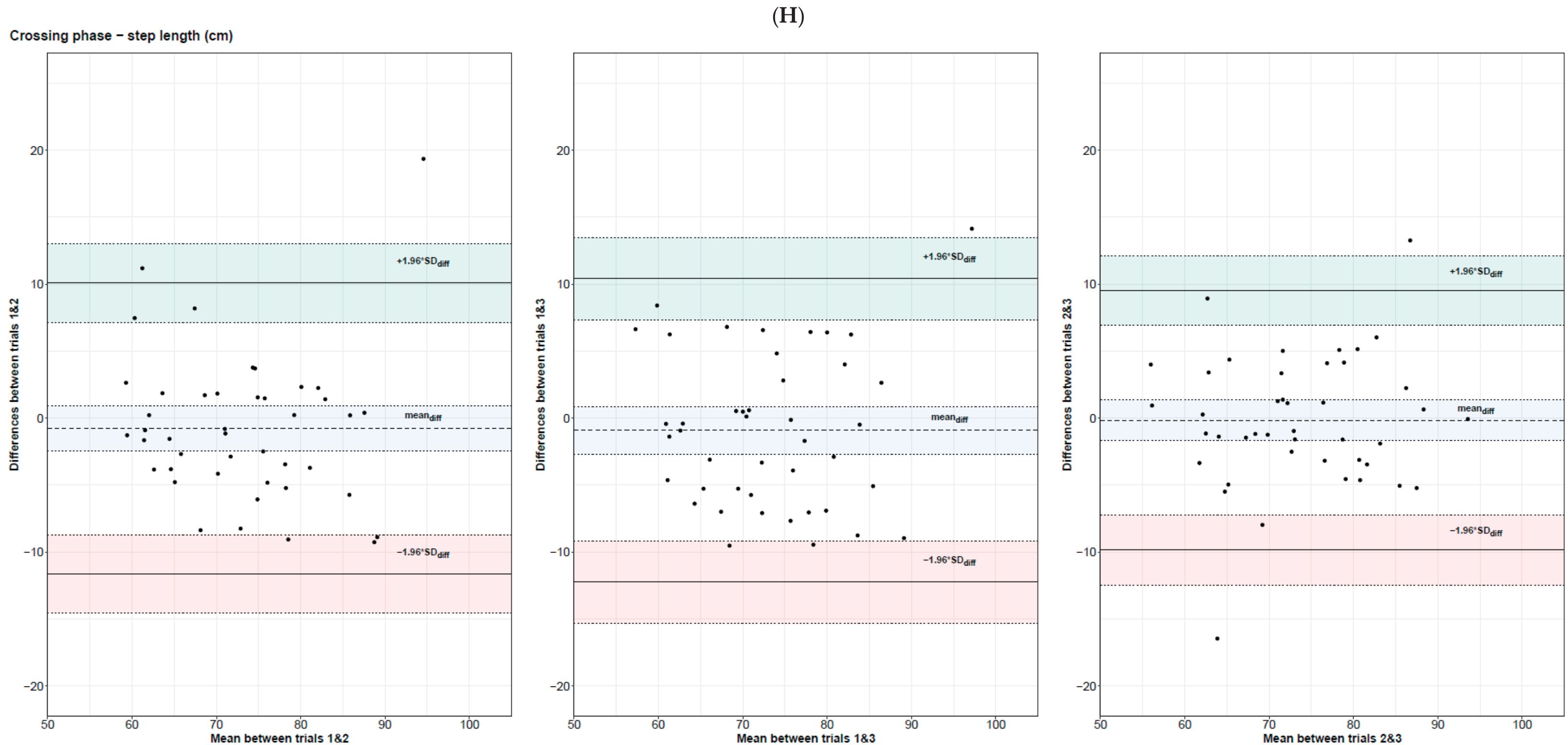


Figure 2. *Cont.*

(I)

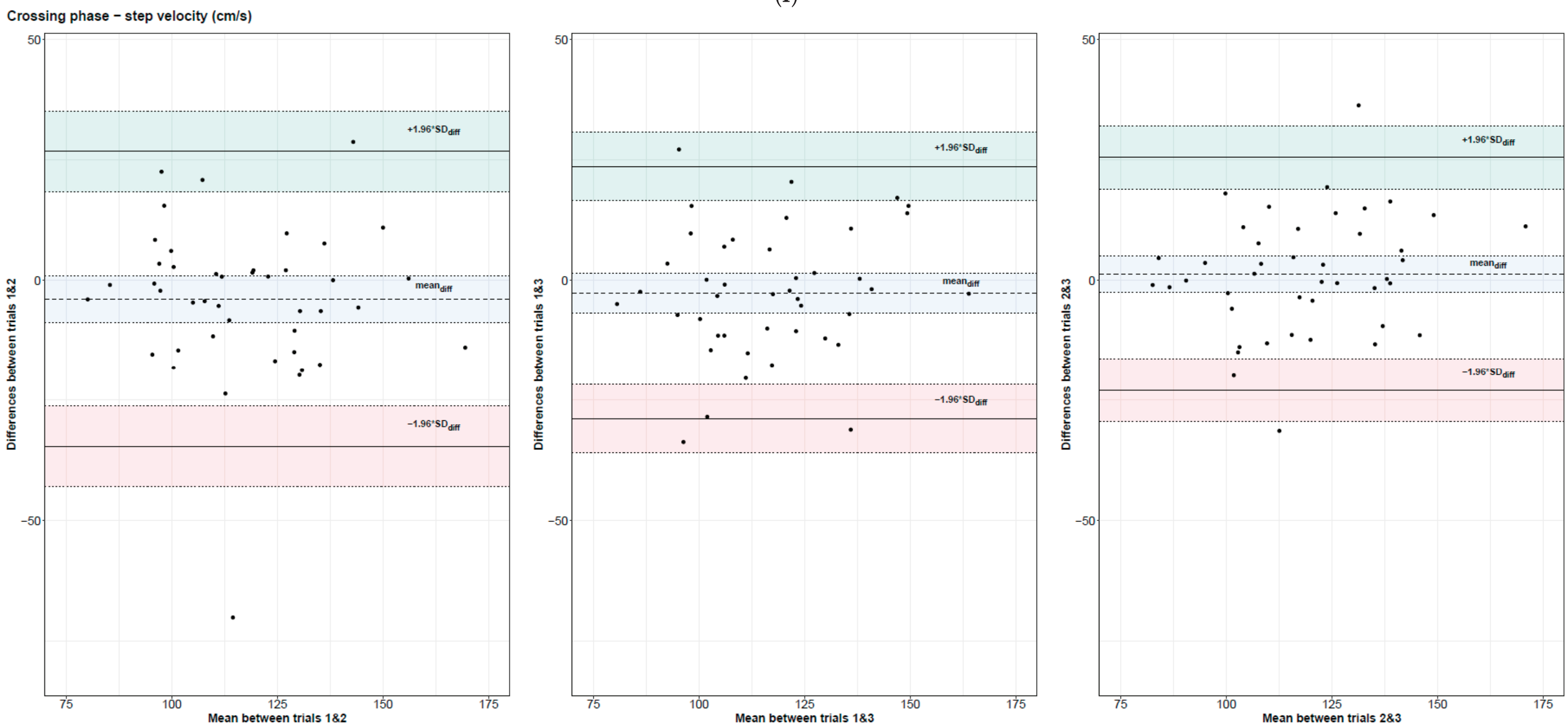


Figure 2. *Cont.*

(J)

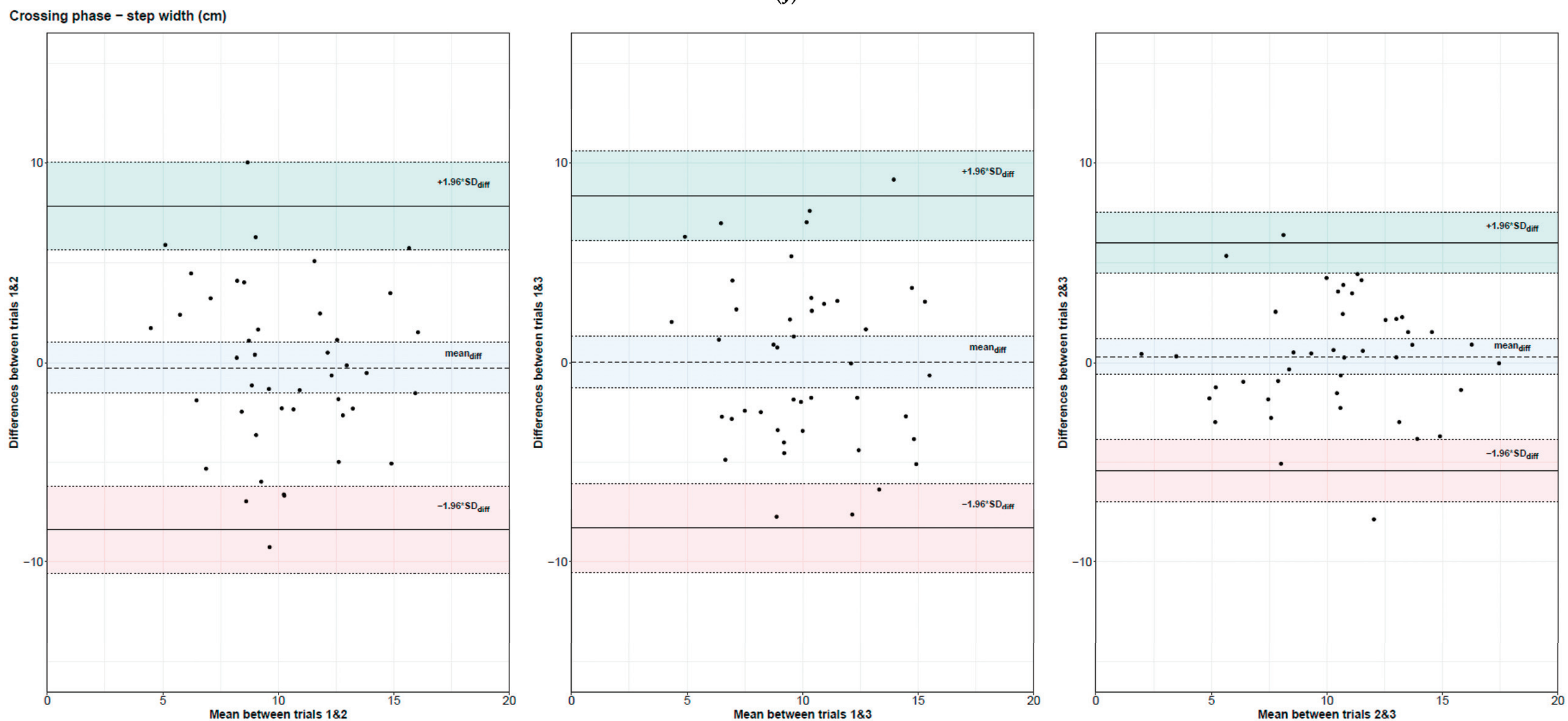


Figure 2. *Cont.*

(**K**)

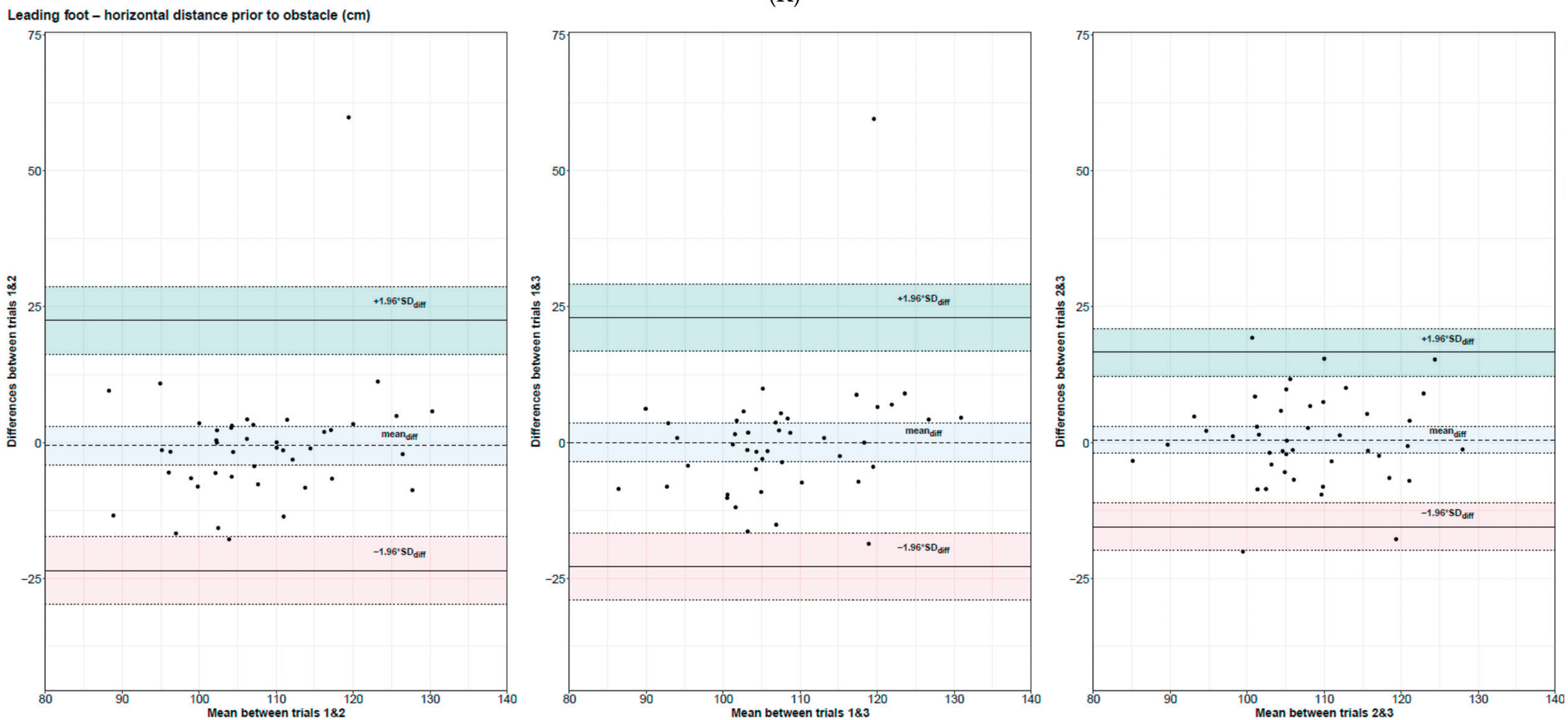


Figure 2. *Cont.*

(L)

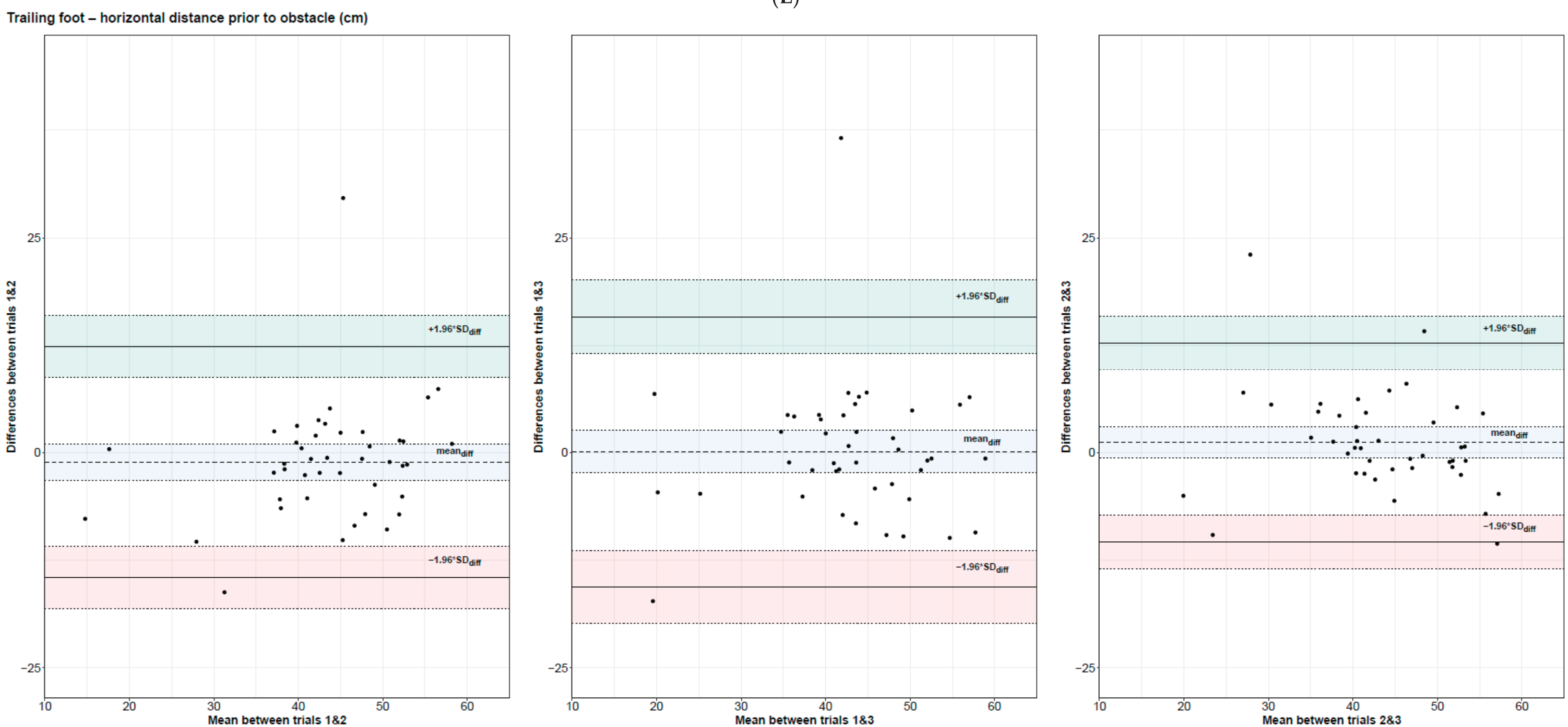


Figure 2. *Cont.*

(M)

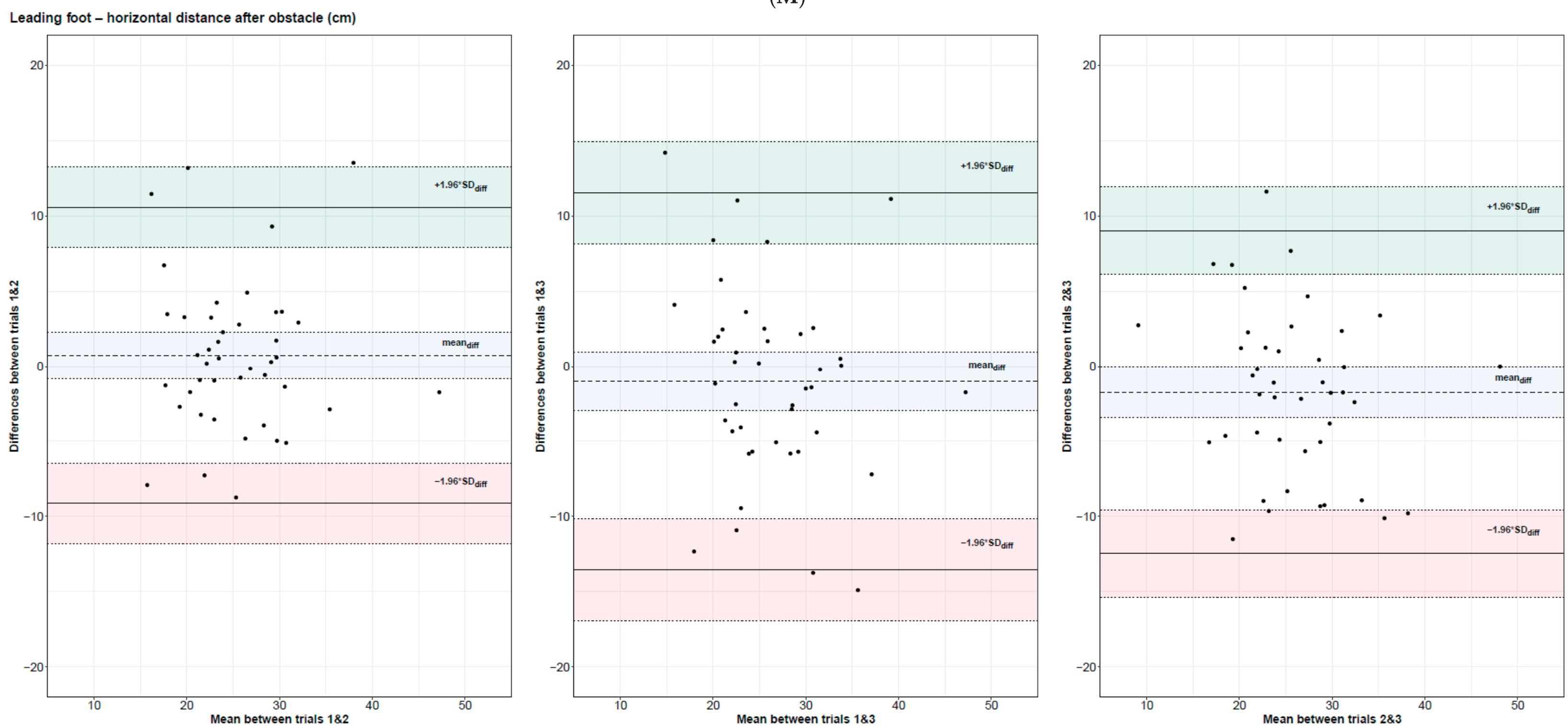


Figure 2. *Cont.*

(N)

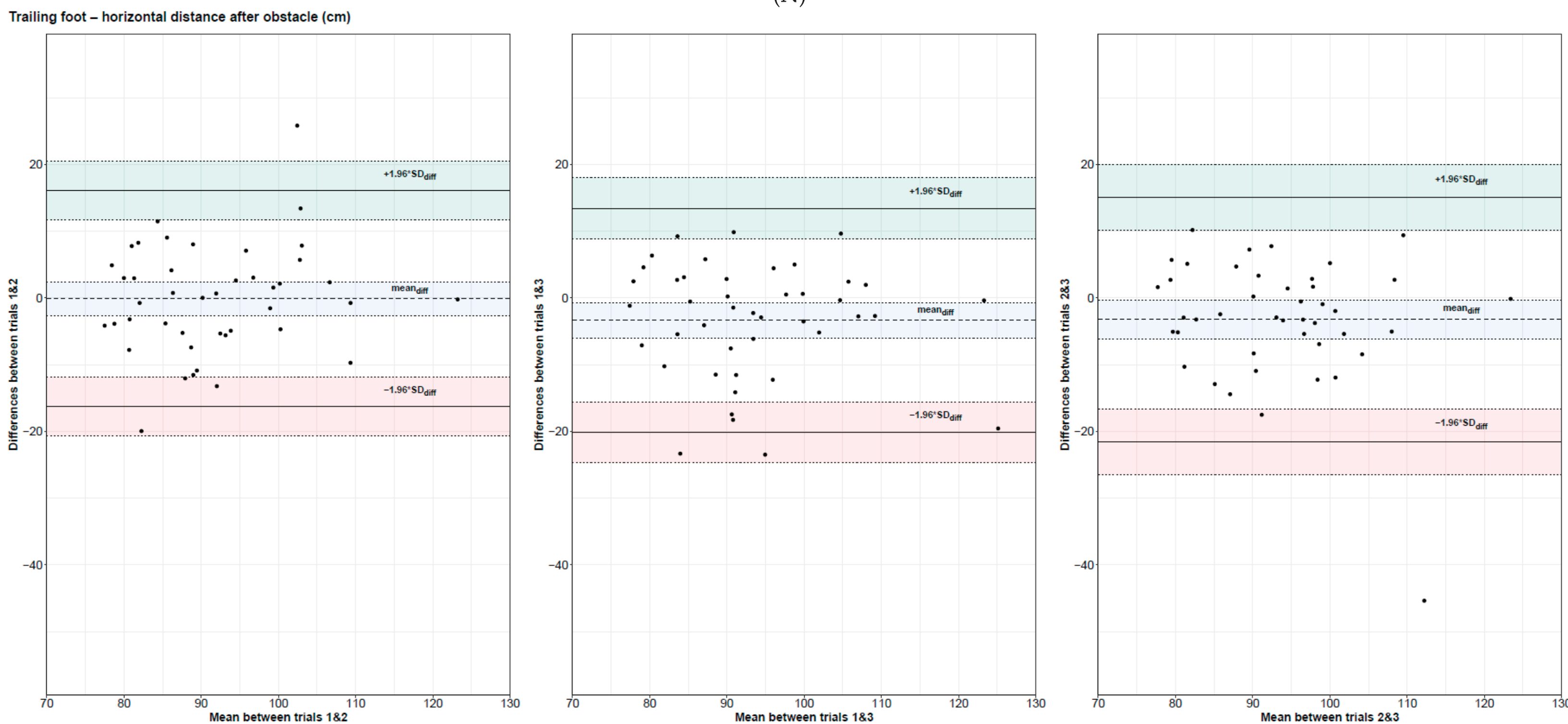


Figure 2. *Cont.*

(O)

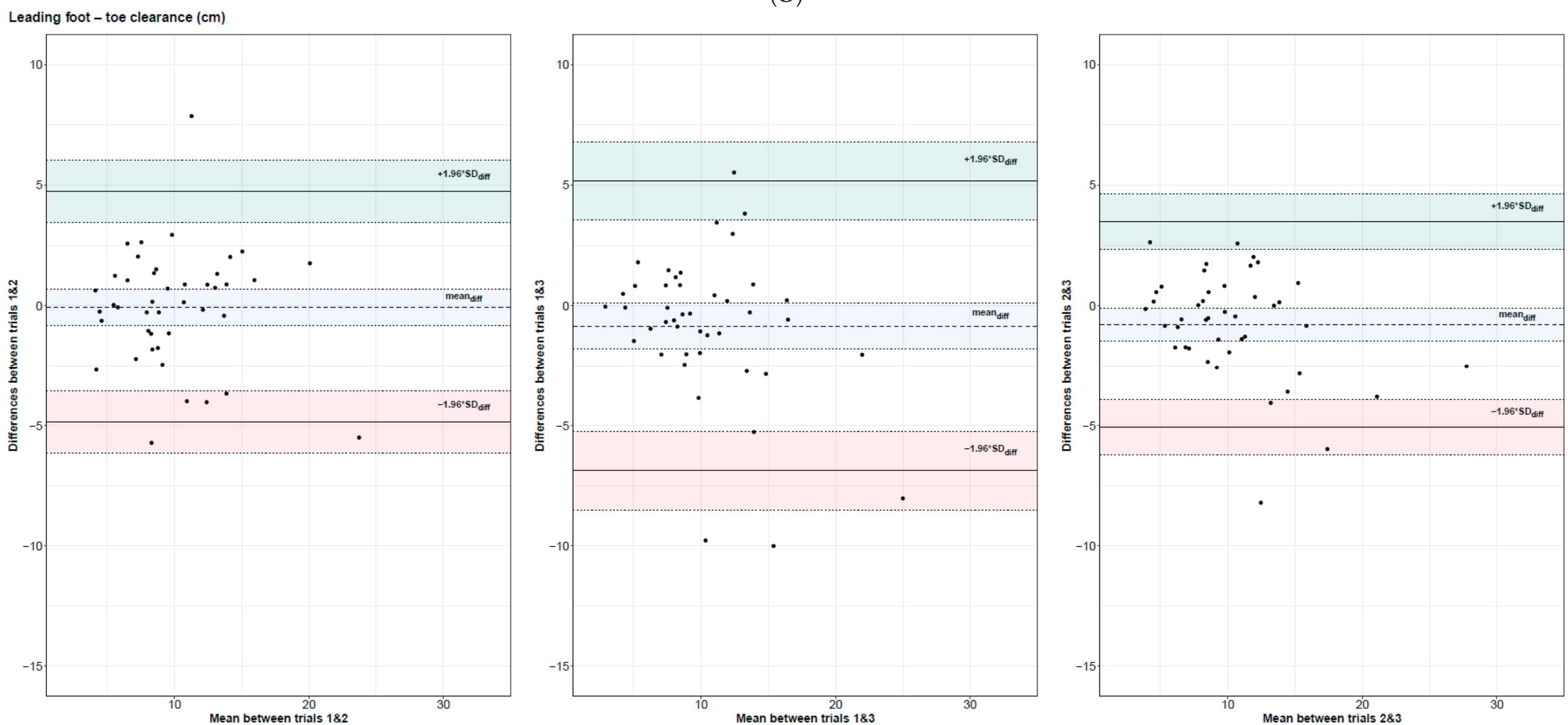


Figure 2. *Cont.*

(**P**)

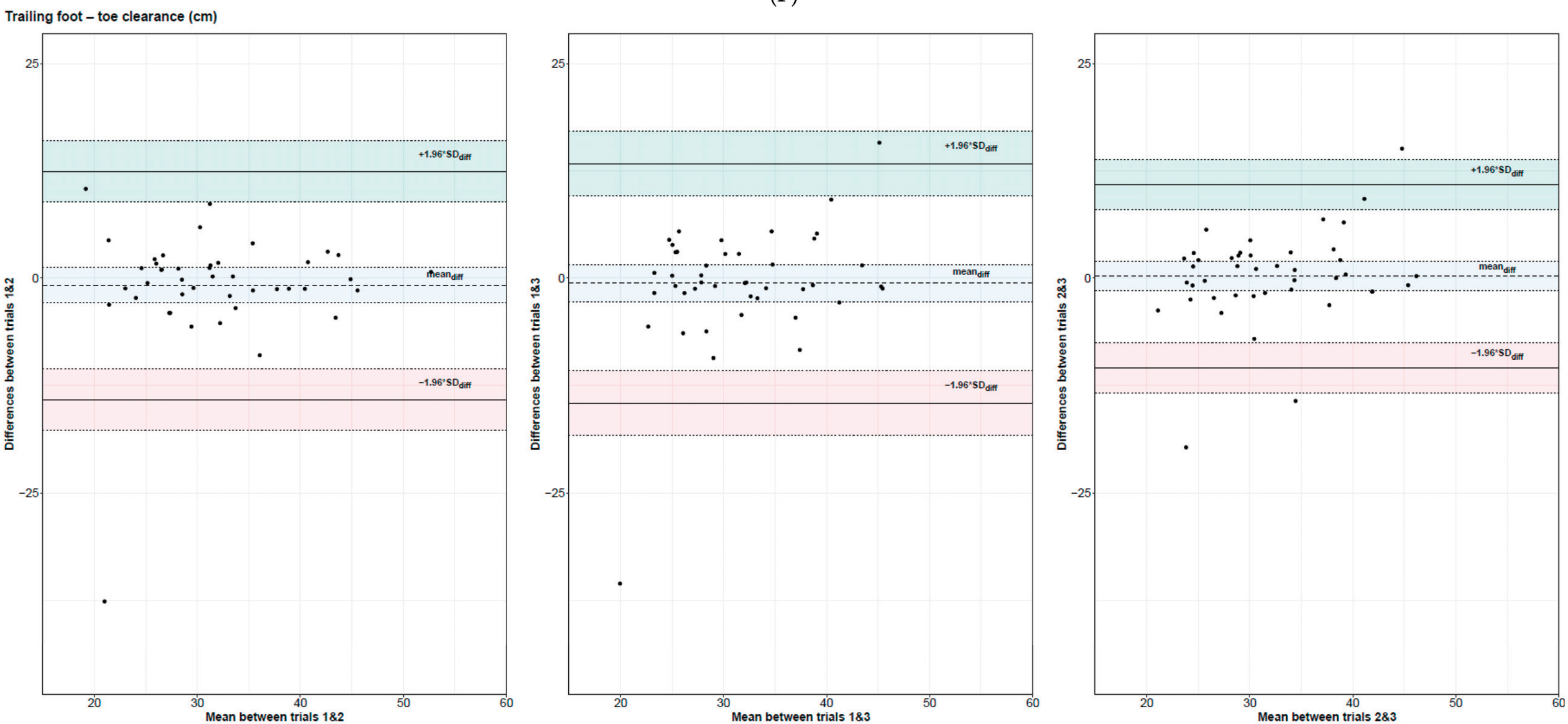


Figure 2. Bland–Altman plots for trial 1 and 2, 1 and 3, and 2 and 3 for gait parameters during the approaching phase (**A**–**E**), the crossing phase (**F**–**J**), and obstacle clearance parameters (**K**–**P**).

4. Discussion

The aim of this study was to evaluate the intra-session relative and absolute reliability of obstacle-crossing parameters during overground walking in young adults and to determine the number of trials required to ensure reliable assessment.

Overall, our results showed for the first time that spatial–temporal gait parameters and clearance parameters were reliable, except for the step width (between T1–T2 and T1–T3) and double support time (between T1–T2) during the crossing phase. In addition, for most spatial–temporal gait and clearance parameters, the reliability and agreement were better when pooling trials 2 and 3 than when the first trial was used for comparison. This suggests that when conducting an obstacle-crossing task in young adults—under the specific experimental conditions used in the present study, at least—a first practice trial should be performed before the experimental trials to enhance the reliability of the parameters studied, or that the first experimental trial should be discarded from data analysis.

4.1. Intra-Session Reliability during Obstacle-Crossing

4.1.1. Approaching Phase

Relative reliability for all spatial–temporal gait parameters measured in young adults during the approaching phase was always either substantial or almost perfect. Regarding absolute reliability, all parameters except step width (SEM% $\geq$ 10%) exhibited good absolute reliability in all trial comparisons, and MDC% was low or acceptable for all gait parameters in all trial comparisons. Comparisons between trials showed that participants had a higher step width in the last two trials compared to the first one. Perhaps after the first trial, the strategy chosen was to increase their pre-crossing base of support after the trial to ensure more stability before the crossing phase. Regarding agreements between trials, results were considered good for all spatial–temporal gait parameters in all trial comparisons. Hence, young male adults had almost identical gait patterns during all trials during the approaching phase. In addition, for all spatial–temporal gait parameters, relative and absolute reliabilities and agreement between trials were better when using the mean of the second and third trials, compared to the first and second trials, to the first and third trials, or to the three trials. This could suggest that participants adapted a more stable/consistent gait pattern after the execution of the first trial, which could have been used as a learning/warming-up trial. These results are similar to those recently reported in healthy adults during unobstructed level walking [2]. In that study, 20 healthy participants performed three trials of a 10-m walking test, and spatial–temporal gait parameters were computed using inertial motor units (Physilog® 5, 200Hz GaitUp, Lausanne, Switzerland). Results showed that speed, double support, and stride length had almost perfect relative reliability (0.81 < ICC(2.1)). In addition, absolute reliability was good, with low SEM% (1.74% $\leq$ SEM% $\leq$ 6.58% in all trial comparisons) and low MDC% (4.82% $\leq$ MDC% $\leq$ 18.25% for all trial comparisons) for these three spatial–temporal gait parameters [2].

4.1.2. Crossing Phase

Contrary to what was observed for the approaching phase, not all spatial–temporal gait parameters were found to have substantial or almost perfect relative reliability, good absolute reliability, or good agreement between trials.

The relative reliability of step length, duration, and velocity was substantial to almost perfect regardless of the trial comparisons considered. Conversely, step width and double support time showed slight to moderate relative reliability (0.19 $\leq$ ICC(2.1) $\leq$ 0.47), with the worst results when the first trial was included. Regarding absolute reliability, both step width and double support time showed poor reliability in all trial comparisons, except the MDC% of double support time, which was acceptable for the T2–T3 comparison (MDC% = 34.4). In addition, agreements between trials were poor for step width (T1–T2 and T1–T3) and double support time (T1–T2), compared to the other spatial–temporal parameters (moderate or good) between each pair of trials. Comparison between trials showed no differences in step width or double support time during the crossing phase ($p < 0.05$). One hypothesis is that the lower reliability found for step width and double support time parameters during the crossing phase compared to the approaching phase could be explained by the more unusual nature of this step compared to the approaching phase (i.e., walking on flat ground) and due to the higher demand of the task in neuromuscular [62] and cognitive resources [63] than flat ground walking. Also, medial lateral dynamic balance during the crossing phase might be harder to maintain, so it is difficult for the participants to maintain a stable behaviour for crossing step width. However, it is worth noting that agreements between trials were better for all spatial–temporal gait parameters, with the exception of step duration (for which the best value was obtained for T1–T3) when pooling values from the second and third trials.

4.1.3. Obstacle Clearance

Relative reliability for all clearance parameters measured in young adults was either moderate to almost perfect ($0.51 \leq$ ICC(2.1) ≤ 0.89), and the reliability was enhanced from substantial to almost perfect when pooling the second and third trials ($0.63 \leq$ ICC(2.1) ≤ 0.89).

However, the absolute reliability was low for four clearance parameters (leading and trailing foot horizontal distance after obstacle, trailing foot horizontal distance after obstacle, and leading foot toe clearance) for T1–T2, T1–T3, and T1–T2–T3, and three clearance parameters for T2–T3 (leading and trailing foot toe clearance and leading foot horizontal distance after obstacle). In addition, MDC% was high (i.e., 40% $\leq$ MDC%) for two clearance parameters for T1–T2 (leading and trailing foot clearance), T2–T3 (leading foot toe clearance and horizontal distance after obstacle), and T1–T2–T3 (leading foot toe clearance and horizontal distance after obstacle), and for three clearance parameters for T1–T3 (leading and trailing foot toe clearance and leading foot horizontal distance after obstacle).

On the one hand, in general, clearance parameters were less reliable than spatial–temporal gait parameters measured during the approaching and crossing phases (with the exception of stride width and double support time during the crossing phase). As mentioned, the more unusual nature of this step compared to flat ground walking might have played a role in individuals' behaviour and clearance strategy. In addition, it is possible that the higher cognitive load and precise motor control during obstacle-crossing compared to unobstructed walking [63] may contribute to the reduced reliability of clearance parameters. On the other hand, similar to what was observed for spatial–temporal gait parameters computed before and after the obstacle, for most obstacle clearance parameters, both relative and absolute reliabilities were better when using the mean of the second and third trials, compared to the first and second trials, to the first and third trials, or to the three trials. This reinforces the idea of a learning/warming-up effect after the first trial for individuals when they perform an obstacle-crossing task. However, comparisons between trials showed that trailing foot horizontal distance after the obstacle was higher in the third trial compared to the first and second. Together, these results suggest that, contrary to spatial–temporal parameters, individuals might need more than three trials to have reliable behaviour. Regarding agreements between trials, moderate to good agreement was found for all parameters between each pair of trials. Similarly to relative and absolute reliabilities, the best results were found for four out of six clearance parameters (leading and trailing toe clearance and distance prior to obstacle) when pooling the second and third trials. However, agreements between trials were better for T1–T2 for both leading and trailing foot distances after the obstacle.

As shown by measures of agreement (CCC and Bland–Altman, see Figure 2), there should not be any systematic bias in measurements since no obvious relation existed between the difference and the mean of any pair of two trials. In addition, the LOA were relatively small, with few to no outliers depending on the obstacle clearance parameter considered.

Having high absolute reliability (10% < SEM%) indicates that there were intra-individual variabilities between trials.

4.2. Study Limitations, Strengths, and Perspectives

To the authors' best knowledge, this is the first study that assessed the reliability, both relative and absolute, and agreements between trials of spatial–temporal gait parameters and obstacle clearance parameters in young adults. It should be noted that although previous studies have reported either the absolute or relative reliability of these parameters (e.g., [64–66]), none of them addressed our specific research questions. Indeed, we aimed to evaluate the intra-session relative and absolute reliability of obstacle-crossing parameters during overground walking in young healthy adults, and to determine the number of trials required to ensure reliable assessment. Contrary to us, the study from Punt et al. (2017) assessed virtual obstacle-crossing task during treadmill walking in stroke survivors [64], while Grinberg et al. (2021) did not assess obstacle-crossing parameters during overground walking [66], and Said et al. (2009) did not involve young healthy adults [65]. In addition, the sample size of the present study is relatively high (n = 43), compared to previously conducted gait reliability studies [1–3,5,8,11,38]. At this point, it is important to mention that, fully in line with the scientific roadmap we have set up to assess the intra-session reliability of the 10 m walk test [2], we focused on young healthy adults as a first step to provide normative reference values for a young healthy population (e.g., see [67–69]). Naturally, the reliability must now be further tested in other populations with different socio-demographic, anthropometric, clinical, and lifestyle characteristics, including individuals who are overweight or obese (e.g., see [70] for a review), children, adolescents, middle age, and the elderly (e.g., see [71] for a review), and pathological populations (e.g., see [72–74] for reviews) who may also present different levels of fatigue (e.g., see [75] for a review) or physical fitness (e.g., see [76] for a review).

Finally, the present findings highlighted that most spatial–temporal and obstacle clearance parameters were reliable across the three trials and that better relative and absolute reliabilities, as well as agreements between trials, were found when pooling the second and third trials. Nevertheless, several limitations warrant further consideration and caution regarding the interpretation of the results.

The reliability and agreement were assessed during a single experimental session, which prevented us from studying the inter-session reliability. In addition, only young healthy males aged between 20–40 years old were included, and only one experimental condition (15 cm height obstacle; self-selected walking speed; single-task condition) was assessed. In other words, strictly speaking, our conclusion applies only to the population tested and the experimental conditions used in the present study. These experimental conditions could be considered as relatively mild and easy in that the task consisted of crossing a physical obstacle 15 centimetres high at a comfortable speed without any other particular constraint. Further works are thus needed to generalize them in more realistic conditions. For instance, it could be interesting to replicate this reliability study while walking over different and challenging obstacle paradigms that may predispose the participants to an increased risk of foot contact with the obstacle [77–79] and/or with individuals with balance and gait disorders for whom the consequences of imbalance and tripping over obstacles could be more dramatic [80,81]. Hence, these specific issues deserve investigations, which are included in our immediate plans. Another potential future investigation would be to perform more than three trials of obstacle-crossing. This would probably enhance the reliability of all parameters, especially the clearances with the obstacle. This might enable us to determine the minimum number of trials to be carried out, and the number of trials from which data analysis should begin in order to obtain reliable and valid data.

5. Conclusions

Most spatial–temporal gait parameters and obstacle clearance parameters computed using a three-dimensional motion analysis system in young adults during an obstacle-crossing task are reliable using the average of three trials. Our findings further suggest using the mean of the second and third trials to ensure the best relative and absolute reliabilities of most obstacle-crossing parameters. However, our findings only apply to the population tested and to the relatively mild and easy experimental conditions used in the present study. Further works are thus needed to generalize them in more realistic conditions and in other populations.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/s24113387/s1. S1. List of R packages used for statistical analysis [82–87].

Author Contributions: Conceptualization: M.C., F.A.B., P.P. and N.V.; formal analysis: M.C., F.A.B., P.P. and N.V.; funding acquisition: F.A.B. and N.V.; methodology: M.C., F.A.B., P.P. and N.V.; project administration: F.A.B. and N.V.; supervision: F.A.B. and N.V.; visualization: P.P.; writing (original draft): M.C.; writing (review and editing): M.C., F.A.B., P.P. and N.V. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the French National Research Agency (ANR) in the framework of the Investissements d'avenir program (ANR-10-AIRT-05 and ANR-15-IDEX-02), the MIAI @ Grenoble Alpes (ANR-19-P3IA-0003), the São Paulo Research Foundation (FAPESP) (grant #18/18078-0), the National Council for Scientific and Technological Development—CNPq, and in part by the Brazil (CAPES)—Finance Code 001.

Institutional Review Board Statement: This study was approved by the University Institution Review Board (authorization number: CAAE#99191318.0.0000.5398). The study and all methods were performed in accordance with the Declaration of Helsinki.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Acknowledgments: This work is part of the Ph.D. thesis of the first author (M.C.) at Université Grenoble Alpes, France, and São Paulo State University, Bauru, Brazil. The sponsors were not involved in the review or approval of the manuscript for publication. This work also forms part of a broader translational and interdisciplinary research program GaitAlps (N.V.).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wilken, J.M.; Rodriguez, K.M.; Brawner, M.; Darter, B.J. Reliability and Minimal Detectible Change Values for Gait Kinematics and Kinetics in Healthy Adults. *Gait Posture* **2012**, *35*, 301–307. [CrossRef]
2. Soulard, J.; Vaillant, J.; Balaguier, R.; Vuillerme, N. Spatio-Temporal Gait Parameters Obtained from Foot-Worn Inertial Sensors Are Reliable in Healthy Adults in Single- and Dual-Task Conditions. *Sci. Rep.* **2021**, *11*, 10229. [CrossRef] [PubMed]
3. Couto, A.G.B.; Vaz, M.A.P.; Pinho, L.; Félix, J.; Moreira, J.; Pinho, F.; Mesquita, I.A.; Montes, A.M.; Crasto, C.; Sousa, A.S.P. Repeatability and Temporal Consistency of Lower Limb Biomechanical Variables Expressing Interlimb Coordination during the Double-Support Phase in People with and without Stroke Sequelae. *Sensors* **2023**, *23*, 2526. [CrossRef] [PubMed]
4. Giannouli, E.; Kim, E.K.; Fu, C.; Weibel, R.; Sofios, A.; Infanger, D.; Portegijs, E.; Rantanen, T.; Huang, H.; Schmidt-Trucksäss, A.; et al. Psychometric Properties of the MOBITEC-GP Mobile Application for Real-Life Mobility Assessment in Older Adults. *Geriatr. Nurs.* **2022**, *48*, 273–279. [CrossRef]
5. Olsen, S.; Rashid, U.; Allerby, C.; Brown, E.; Leyser, M.; McDonnell, G.; Alder, G.; Barbado, D.; Shaikh, N.; Lord, S.; et al. Smartphone-Based Gait and Balance Accelerometry Is Sensitive to Age and Correlates with Clinical and Kinematic Data. *Gait Posture* **2023**, *100*, 57–64. [CrossRef] [PubMed]
6. Yavuzer, G.; Öken, Ö.; Elhan, A.; Stam, H.J. Repeatability of Lower Limb Three-Dimensional Kinematics in Patients with Stroke. *Gait Posture* **2008**, *27*, 31–35. [CrossRef] [PubMed]
7. Lefeber, N.; Degelaen, M.; Truyers, C.; Safin, I.; Beckwee, D. Validity and Reproducibility of Inertial Physilog Sensors for Spatiotemporal Gait Analysis in Patients with Stroke. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2019**, *27*, 1865–1874. [CrossRef] [PubMed]
8. Wedege, P.; Steffen, K.; Strøm, V.; Opheim, A.I. Reliability of Three-Dimensional Kinematic Gait Data in Adults with Spinal Cord Injury. *J. Rehabil. Assist. Technol. Eng.* **2017**, *4*, 205566831772999. [CrossRef]
9. Fransen, M.; Crosbie, J.; Edmonds, J. Reliability of Gait Measurements in People with Osteoarthritis of the Knee. *Phys. Ther.* **1997**, *77*, 944–953. [CrossRef]
10. Hansen, C.; Ortlieb, C.; Romijnders, R.; Warmerdam, E.; Welzel, J.; Geritz, J.; Maetzler, W. Reliability of IMU-Derived Temporal Gait Parameters in Neurological Diseases. *Sensors* **2022**, *22*, 2304. [CrossRef]

11. Allet, L.; Armand, S.; de Bie, R.A.; Golay, A.; Monnin, D.; Aminian, K.; de Bruin, E.D. Reliability of Diabetic Patients' Gait Parameters in a Challenging Environment. *Gait Posture* **2008**, *28*, 680–686. [CrossRef] [PubMed]
12. Brach, J.S.; Perera, S.; Studenski, S.; Newman, A.B. The Reliability and Validity of Measures of Gait Variability in Community-Dwelling Older Adults. *Arch. Phys. Med. Rehabil.* **2008**, *89*, 2293–2296. [CrossRef]
13. Berard, J.R.; Vallis, L.A. Characteristics of Single and Double Obstacle Avoidance Strategies: A Comparison between Adults and Children. *Exp. Brain Res.* **2006**, *175*, 21–31. [CrossRef] [PubMed]
14. Yoshimoto, K.; Mani, H.; Hirose, N.; Kurogi, T.; Aiko, T.; Shinya, M. Dynamic Stability during Level Walking and Obstacle Crossing in Children Aged 2–5 Years Estimated by Marker-Less Motion Capture. *Front. Sports Act. Living* **2023**, *5*, 1109581. [CrossRef] [PubMed]
15. Pavol, M.J.; Owings, T.M.; Foley, K.T.; Grabiner, M.D. Mechanisms Leading to a Fall from an Induced Trip in Healthy Older Adults. *J. Gerontol. A Biol. Sci. Med. Sci.* **2001**, *56*, M428–M437. [CrossRef]
16. Chen, H.C.; Ashton-Miller, J.A.; Alexander, N.B.; Schultz, A.B. Stepping over Obstacles: Gait Patterns of Healthy Young and Old Adults. *J. Gerontol.* **1991**, *46*, M196–M203. [CrossRef]
17. Gill, S.V.; Hung, Y.C. Effects of Overweight and Obese Body Mass on Motor Planning and Motor Skills during Obstacle Crossing in Children. *Res. Dev. Disabil.* **2014**, *35*, 46–53. [CrossRef]
18. Galna, B.; Murphy, A.T.; Morris, M.E. Obstacle Crossing in Parkinson's Disease: Mediolateral Sway of the Centre of Mass during Level-Ground Walking and Obstacle Crossing. *Gait Posture* **2013**, *38*, 790–794. [CrossRef]
19. dos Santos, P.C.R.; Barbieri, F.A.; Orcioli-Silva, D.; Beretta, V.S.; Hortobágyi, T.; Gobbi, L.T.B. Being Physically Active Minimizes the Effects of Leg Muscle Fatigue on Obstacle Negotiation in People with Parkinson's Disease. *J. Biomech.* **2021**, *124*, 110568. [CrossRef]
20. Penedo, T.; Polastri, P.F.; Rodrigues, S.T.; Simieli, L.; Baptista, A.M.; Moretto, G.F.; Imaizumi, L.F.I.; Santinelli, F.B.; Barbieri, F.A. Influence of Obstacle Color on Locomotor and Gaze Behaviors during Obstacle Avoidance in People with Parkinson's Disease. *Exp. Brain Res.* **2018**, *236*, 3319–3325. [CrossRef]
21. Simieli, L.; Gobbi, L.T.B.; Orcioli-Silva, D.; Beretta, V.S.; Santos, P.C.R.; Baptista, A.M.; Barbieri, F.A. The Variability of the Steps Preceding Obstacle Avoidance (Approach Phase) Is Dependent on the Height of the Obstacle in People with Parkinson's Disease. *PLoS ONE* **2017**, *12*, e0184134. [CrossRef]
22. Santinelli, F.B.; Sebastião, E.; Kuroda, M.H.; Moreno, V.C.; Pilon, J.; Vieira, L.H.P.; Barbieri, F.A. Cortical Activity and Gait Parameter Characteristics in People with Multiple Sclerosis during Unobstructed Gait and Obstacle Avoidance. *Gait Posture* **2021**, *86*, 226–232. [CrossRef]
23. Chou, L.S.; Kaufman, K.R.; Walker-Rabatin, A.E.; Brey, R.H.; Basford, J.R. Dynamic Instability during Obstacle Crossing Following Traumatic Brain Injury. *Gait Posture* **2004**, *20*, 245–254. [CrossRef] [PubMed]
24. Den Otter, A.R.; Geurts, A.C.H.; De Haart, M.; Mulder, T.; Duysens, J. Step Characteristics during Obstacle Avoidance in Hemiplegic Stroke. *Exp. Brain Res.* **2005**, *161*, 180–192. [CrossRef]
25. Said, C.M.; Galea, M.; Lythgo, N. Obstacle Crossing Following Stroke Improves over One Month When the Unaffected Limb Leads, but Not When the Affected Limb Leads. *Gait Posture* **2014**, *39*, 213–217. [CrossRef] [PubMed]
26. Scalvini, F.; Bordeau, C.; Ambard, M.; Migniot, C.; Vergnaud, M.; Dubois, J. UB-VisioGeoloc: An Image Sequences Dataset of Pedestrian Navigation Including Geolocalised-Inertial Information and Spatial Sound Rendering of the Urban Environment's Obstacles. *Data Brief.* **2024**, *53*, 110088. [CrossRef]
27. Soma, M.; Nakae, H.; Abiko, T.; Shimamura, R.; Uematsu, H.; Kawama, K. Influence of a Dual Task While Stepping Over an Obstacle in the Fall-Experienced Elderly People. *J. Phys. Ther. Sci.* **2011**, *23*, 369–372. [CrossRef]
28. Weerdesteyn, V.; Nienhuis, B.; Duysens, J. Advancing Age Progressively Affects Obstacle Avoidance Skills in the Elderly. *Hum. Mov. Sci.* **2005**, *24*, 865–880. [CrossRef] [PubMed]
29. Gök, H.; Ergin, S.; Yavuzer, G. Reliability of Gait Measurements in Normal Subjects. *J. Rheumatol. Med. Rehabil.* **2002**, *13*, 76–80.
30. Boldo, M.; Di Marco, R.; Martini, E.; Nardon, M.; Bertucco, M.; Bombieri, N. On the Reliability of Single-Camera Markerless Systems for Overground Gait Monitoring. *Comput. Biol. Med.* **2024**, *171*, 108101. [CrossRef]
31. van Bloemendaal, M.; Beelen, A.; Kleissen, R.F.M.; Geurts, A.C.H.; Nollet, F.; Bus, S.A. Concurrent Validity and Reliability of a Low-Cost Gait Analysis System for Assessment of Spatiotemporal Gait Parameters. *J. Rehabil. Med.* **2019**, *51*, 456–463. [CrossRef] [PubMed]
32. Widhalm, K.; Durstberger, S.; Greisberger, A.; Wolf, B.; Putz, P. Validity of Assessing Level Walking with the 2D Motion Analysis Software TEMPLO and Reliability of 3D Marker Application. *Sci. Rep.* **2024**, *14*, 1427. [CrossRef]
33. Girard, O.; Brocherie, F.; Morin, J.B.; Millet, G.P. Intrasession and Intersession Reliability of Running Mechanics During Treadmill Sprints. *Int. J. Sports Physiol. Perform.* **2016**, *11*, 432–439. [CrossRef] [PubMed]
34. Pinsault, N.; Vuillerme, N. Test-Retest Reliability of Centre of Foot Pressure Measures to Assess Postural Control during Unperturbed Stance. *Med. Eng. Phys.* **2009**, *31*, 276–286. [CrossRef]
35. Pinsault, N.; Fleury, A.; Virone, G.; Bouvier, B.; Vaillant, J.; Vuillerme, N. Test-Retest Reliability of Cervicocephalic Relocation Test to Neutral Head Position. *Physiother. Theory Pract.* **2008**, *24*, 380–391. [CrossRef]
36. Balaguier, R.; Madeleine, P.; Vuillerme, N. Is One Trial Sufficient to Obtain Excellent Pressure Pain Threshold Reliability in the Low Back of Asymptomatic Individuals? A Test-Retest Study. *PLoS ONE* **2016**, *11*, e0160866. [CrossRef] [PubMed]

37. Balaguier, R.; Madeleine, P.; Vuillerme, N. Intra-Session Absolute and Relative Reliability of Pressure Pain Thresholds in the Low Back Region of Vine-Workers: Effect of the Number of Trials. *BMC Musculoskelet. Disord.* **2016**, *17*, 350. [CrossRef] [PubMed]
38. Soulard, J.; Vaillant, J.; Balaguier, R.; Baillet, A.; Gaudin, P.; Vuillerme, N. Foot-Worn Inertial Sensors Are Reliable to Assess Spatiotemporal Gait Parameters in Axial Spondyloarthritis under Single and Dual Task Walking in Axial Spondyloarthritis. *Sensors* **2020**, *20*, 6453. [CrossRef] [PubMed]
39. Linek, P.; Sikora, D.; Wolny, T.; Saulicz, E. Reliability and Number of Trials of Y Balance Test in Adolescent Athletes. *Musculoskelet. Sci. Pract.* **2017**, *31*, 72–75. [CrossRef]
40. Karatrantou, K.; Xagorari, A.; Vasilopoulou, T.; Gerodimos, V. Does the Number of Trials Affect the Reliability of Handgrip Strength Measurement in Individuals with Intellectual Disabilities? *Hand Surg. Rehabil.* **2020**, *39*, 223–228. [CrossRef]
41. Pineda, R.C.; Krampe, R.T.; Vanlandewijck, Y.; Van Biesen, D. Reliability of Center of Pressure Excursion as a Measure of Postural Control in Bipedal Stance of Individuals with Intellectual Disability: A Pilot Study. *PLoS ONE* **2020**, *15*, e0240702. [CrossRef]
42. Severin, A.C.; Barnes, S.G.; Tackett, S.A.; Barnes, C.L.; Mannen, E.M. The Required Number of Trials for Biomechanical Analysis of a Golf Swing. *Sports Biomech.* **2021**, *20*, 238–246. [CrossRef] [PubMed]
43. Seuthe, J.; D'Cruz, N.; Ginis, P.; Blöbaum, R.; Weisser, B.; Deuschl, G.; Nieuwboer, A.; Schlenstedt, C. How Many Gait Initiation Trials Are Necessary to Reliably Detect Anticipatory Postural Adjustments and First Step Characteristics in Healthy Elderly and People with Parkinson's Disease? *Gait Posture* **2021**, *88*, 126–131. [CrossRef]
44. Silva, A.F.; Nobari, H.; Badicu, G.; Ceylan, H.I.; Lima, R.; Lagoa, M.J.; Luz, C.; Clemente, F.M. Reliability Levels of Motor Competence in Youth Athletes. *BMC Pediatr.* **2022**, *22*, 430. [CrossRef] [PubMed]
45. Coleman, J.L.; Huynh, M.; Middleton, K.J. Number of Trials Necessary to Achieve a Representative Performance of Accuracy and Timing during Combat Shooting. *Scand. J. Med. Sci. Sports* **2023**, *33*, 2039–2045. [CrossRef] [PubMed]
46. Villarrasa-Sapiña, I.; Ortega-Benavent, N.; Monfort-Torres, G.; Ramon-Llin, J.; García-Massó, X. Test-Retest Reliability of Task Performance for Golf Swings of Medium- to High-Handicap Players. *Sensors* **2022**, *22*, 9069. [CrossRef] [PubMed]
47. García-Massó, X.; Villarrasa-Sapiña, I.; Ortega-Benavent, N.; Montalt-García, S.; Toca-Herrera, J.L. Test-Retest Reliability of Putting-Related Variables in Medium-to-High Handicap Golf Players. *Sci. Rep.* **2024**, *14*, 11516. [CrossRef]
48. Monaghan, K.; Delahunt, E.; Caulfield, B. Increasing the Number of Gait Trial Recordings Maximises Intra-Rater Reliability of the CODA Motion Analysis System. *Gait Posture* **2007**, *25*, 303–315. [CrossRef] [PubMed]
49. Santinelli, F.B.; Sebastião, E.; Simieli, L.; Antunes, B.M.; Vieira, L.H.P.; Kalron, A.; Barbieri, F.A. Is BDNF Related to Spatial-Temporal Gait Parameters in People with Multiple Sclerosis? An Observational Study. *Mult. Scler. Relat. Disord.* **2022**, *66*, 104064. [CrossRef]
50. Barbieri, F.A.; dos Santos, P.C.R.; Simieli, L.; Orcioli-Silva, D.; van Dieën, J.H.; Gobbi, L.T.B. Interactions of Age and Leg Muscle Fatigue on Unobstructed Walking and Obstacle Crossing. *Gait Posture* **2014**, *39*, 985–990. [CrossRef]
51. Landis, J.; Koch, G. The Measurement of Observer Agreement for Categorical Data. *Biometrics* **1977**, *33*, 159–174. [CrossRef] [PubMed]
52. Harvill, L.M. Standard Error of Measurement. *Educ. Meas. Issues Pract.* **1991**, *10*, 33–41. [CrossRef]
53. Walton, D.; Macdermid, J.; Nielson, W.; Teasell, R.; Chiasson, M.; Brown, L. Reliability, Standard Error, and Minimum Detectable Change of Clinical Pressure Pain Threshold Testing in People with and without Acute Neck Pain. *J. Orthop. Sports Phys. Ther.* **2011**, *41*, 644–650. [CrossRef] [PubMed]
54. Weir, J.P. Quantifying Test-Retest Reliability Using the Intraclass Correlation Coefficient and the SEM. *J. Strength. Cond. Res.* **2005**, *19*, 231–240. [CrossRef] [PubMed]
55. Bourke, A.K.; Scotland, A.; Lipsmeier, F.; Gossens, C.; Lindemann, M. Gait Characteristics Harvested During a Smartphone-Based Self-Administered 2-Minute Walk Test in People with Multiple Sclerosis: Test-Retest Reliability and Minimum Detectable Change. *Sensors* **2020**, *20*, 5906. [CrossRef]
56. Huang, S.L.; Hsieh, C.L.; Wu, R.M.; Lu, W.S. Test-Retest Reliability and Minimal Detectable Change of the Beck Depression Inventory and the Taiwan Geriatric Depression Scale in Patients with Parkinson's Disease. *PLoS ONE* **2017**, *12*, e0184823. [CrossRef]
57. Feng, D.; Baumgartner, R.; Svetnik, V. A Short Note on Jackknifing the Concordance Correlation Coefficient. *Stat. Med.* **2014**, *33*, 514–516. [CrossRef] [PubMed]
58. Gillinov, S.; Etiwy, M.; Wang, R.; Blackburn, G.; Phelan, D.; Gillinov, A.M.; Houghtaling, P.; Javadikasgari, H.; Desai, M.Y. Variable Accuracy of Wearable Heart Rate Monitors during Aerobic Exercise. *Med. Sci. Sports Exerc.* **2017**, *49*, 1697–1703. [CrossRef]
59. Shieh, G. Assessing Individual Equivalence in Parallel Group and Crossover Designs: Exact Test and Sample Size Procedures. *PLoS ONE* **2022**, *17*, e0269128. [CrossRef]
60. Quinn, C.; Haber, M.J.; Pan, Y. Use of the Concordance Correlation Coefficient When Examining Agreement in Dyadic Research. *Nurs. Res.* **2009**, *58*, 368–373. [CrossRef]
61. Buchheit, M. Magnitudes Matter More than Beetroot Juice. *Sport Perform. Sci. Rep.* **2018**, *1*, 1–3.
62. MacLellan, M.J.; McFadyen, B.J. Proximal Lower Limb Muscle Energetics and the Adaptation of Segment Elevation Angle Phasing for Obstacle Avoidance. *Gait Posture* **2013**, *37*, 274–279. [CrossRef] [PubMed]
63. Billington, J.; Wilkie, R.M.; Wann, J.P. Obstacle Avoidance and Smooth Trajectory Control: Neural Areas Highlighted during Improved Locomotor Performance. *Front. Behav. Neurosci.* **2013**, *7*, 9. [CrossRef] [PubMed]

64. Punt, M.; Bruijn, S.M.; Wittink, H.; van de Port, I.G.; Wubbels, G.; van Dieën, J.H. Virtual Obstacle Crossing: Reliability and Differences in Stroke Survivors Who Prospectively Experienced Falls or No Falls. *Gait Posture* **2017**, *58*, 533–538. [CrossRef]
65. Said, C.M.; Galea, M.; Lythgo, N. Obstacle Crossing Performance Does Not Differ between the First and Subsequent Attempts in People with Stroke. *Gait Posture* **2009**, *30*, 455–458. [CrossRef] [PubMed]
66. Grinberg, A.; Strong, A.; Buck, S.; Selling, J.; Häger, C.K. An Obstacle Clearance Test for Evaluating Sensorimotor Control after Anterior Cruciate Ligament Injury: A Kinematic Analysis. *J. Orthop. Res.* **2022**, *40*, 105–116. [CrossRef] [PubMed]
67. Wu, A.R.; Simpson, C.S.; van Asseldonk, E.H.F.; van der Kooij, H.; Ijspeert, A.J. Mechanics of Very Slow Human Walking. *Sci. Rep.* **2019**, *9*, 18079. [CrossRef] [PubMed]
68. Lordall, J.; Bruno, P.; Ryan, N. Assessment of Diurnal Variation of Stride Time Variability during Continuous, Overground Walking in Healthy Young Adults. *Gait Posture* **2020**, *79*, 108–110. [CrossRef] [PubMed]
69. Ryan, N.S.; Bruno, P.A.; Barden, J.M. Test-Retest Reliability and the Effects of Walking Speed on Stride Time Variability During Continuous, Overground Walking in Healthy Young Adults. *J. Appl. Biomech.* **2021**, *37*, 102–108. [CrossRef]
70. Chardon, M.; Barbieri, F.A.; Penedo, T.; Santos, P.C.R.; Vuillerme, N. A Systematic Review of the Influence of Overweight and Obesity across the Lifespan on Obstacle Crossing during Walking. *Int. J. Environ. Res. Public. Health* **2023**, *20*, 5931. [CrossRef]
71. Galna, B.; Peters, A.; Murphy, A.T.; Morris, M.E. Obstacle Crossing Deficits in Older Adults: A Systematic Review. *Gait Posture* **2009**, *30*, 270–275. [CrossRef] [PubMed]
72. Muroi, D.; Ohtera, S.; Kataoka, Y.; Banno, M.; Tsujimoto, Y.; Tsujimoto, H.; Higuchi, T. Obstacle Avoidance Training for Individuals with Stroke: A Systematic Review and Meta-Analysis. *BMJ Open* **2019**, *9*, e028873. [CrossRef] [PubMed]
73. Cloutier, M.; Delucia, P.R. Topical Review: Impact of Central Vision Loss on Navigation and Obstacle Avoidance While Walking. *Optom. Vis. Sci.* **2022**, *99*, 890–899. [CrossRef] [PubMed]
74. Caparrós-Manosalva, C.; Espinoza, J.; Caballero, P.M.; da Cunha, M.J.; Yang, F.; Galen, S.; Pagnussat, A.S. Movement Strategies during Obstacle Crossing in People with Parkinson Disease: A Systematic Review with Meta-Analysis. *PM R* **2024**, 1–14. [CrossRef] [PubMed]
75. Chardon, M.; Barbieri, F.A.; Penedo, T.; Santos, P.C.R.; Vuillerme, N. The Effects of Experimentally-Induced Fatigue on Gait Parameters during Obstacle Crossing: A Systematic Review. *Neurosci. Biobehav. Rev.* **2022**, *142*, 104854. [CrossRef] [PubMed]
76. Guadagnin, E.C.; da Rocha, E.S.; Duysens, J.; Carpes, F.P. Does Physical Exercise Improve Obstacle Negotiation in the Elderly? A Systematic Review. *Arch. Gerontol. Geriatr.* **2016**, *64*, 138–145. [CrossRef]
77. Muir, B.C.; Haddad, J.M.; Heijnen, M.J.H.; Rietdyk, S. Proactive Gait Strategies to Mitigate Risk of Obstacle Contact Are More Prevalent with Advancing Age. *Gait Posture* **2015**, *41*, 233–239. [CrossRef] [PubMed]
78. Chen, S.H.; Lo, O.Y.; Kay, T.; Chou, L.S. Concurrent Phone Texting Alters Crossing Behavior and Induces Gait Imbalance during Obstacle Crossing. *Gait Posture* **2018**, *62*, 422–425. [CrossRef] [PubMed]
79. Raffegeau, T.E.; Brinkerhoff, S.A.; Kellaher, G.K.; Baudendistel, S.; Terza, M.J.; Roper, J.A.; Hass, C.J. Changes to Margins of Stability from Walking to Obstacle Crossing in Older Adults While Walking Fast and with a Dual-Task. *Exp. Gerontol.* **2022**, *161*, 111710. [CrossRef]
80. Fingerhut, L.A.; Cox, C.; Warner, M. International Comparative Analysis of Injury Mortality. Findings from the ICE on Injury Statistics. International Collaborative Effort on Injury Statistics. *Adv. Data* **1998**, *303*, 1–20.
81. Voss, T.S.; Elm, J.J.; Wielinski, C.L.; Aminoff, M.J.; Bandyopadhyay, D.; Chou, K.L.; Sudarsky, L.R.; Tilley, B.C. Fall Frequency and Risk Assessment in Early Parkinson's Disease. *Parkinsonism Relat. Disord.* **2012**, *18*, 837–841. [CrossRef] [PubMed]
82. Wickham, H.; Averick, M.; Bryan, J.; Chang, W.; D', L.; Mcgowan, A.; François, R.; Grolemund, G.; Hayes, A.; Henry, L.; et al. Welcome to the Tidyverse. *J. Open Source Softw.* **2019**, *4*, 1686. [CrossRef]
83. Kassambara, A. Pipe-Friendly Framework for Basic Statistical Tests [R Package Rstatix Version 0.7.2]. 2023.
84. Revelle, W. Procedures for Psychological, Psychometric, and Personality Research [R Package Psych Version 2.4.3]. 2024.
85. Huebner, A.; McGinn, T.; Sisk, M. psr: Functions for Analyzing Performance Science Data. 2021.
86. Caldwell, A. SimplyAgree: An R Package and Jamovi Module for Simplifying Agreement and Reliability Analyses. *J. Open Source Softw.* **2022**, *7*, 4148. [CrossRef]
87. Datta, D. blandr: A Bland-Altman Method Comparison package for R. Zenodo. 2017.

sensors

MDPI

Article

Using Inertial Measurement Units to Examine Selected Joint Kinematics in a Road Cycling Sprint: A Series of Single Cases

Simon Morbey, Marius Tronslien, Kunho Kong, Dale W. Chapman and Kevin Netto *

Curtin School of Allied Health, Curtin University, Kent Street, Bentley, WA 6102, Australia; simon.r.morbey@gmail.com (S.M.); mtronslien@gmail.com (M.T.); cedric.kong93@gmail.com (K.K.); dale.chapman@curtin.edu.au (D.W.C.)
* Correspondence: kevin.netto@curtin.edu.au

Abstract: Sprinting plays a significant role in determining the results of road cycling races worldwide. However, currently, there is a lack of systematic research into the kinematics of sprint cycling, especially in an outdoor, environmentally valid setting. This study aimed to describe selected joint kinematics during a cycling sprint outdoors. Three participants were recorded sprinting over 60 meters in both standing and seated sprinting positions on an outdoor course with a baseline condition of seated cycling at 20 km/h. The participants were recorded using array-based inertial measurement units to collect joint excursions of the upper and lower limbs including the trunk. A high-rate GPS unit was used to record velocity during each recorded condition. Kinematic data were analyzed in a similar fashion to running gait, where multiple pedal strokes were identified, delineated, and averaged to form a representative (average ± SD) waveform. Participants maintained stable kinematics in most joints studied during the baseline condition, but variations in ranges of movement were recorded during seated and standing sprinting. Discernable patterns started to emerge for several kinematic profiles during standing sprinting. Alternate sprinting strategies emerged between participants and bilateral asymmetries were also recorded in the individuals tested. This approach to studying road cycling holds substantial potential for researchers wishing to explore this sport.

Keywords: sprint cycling; road cycling; inertial measurement units; kinematics; elite

Citation: Morbey, S.; Tronslien, M.; Kong, K.; Chapman, D.W.; Netto, K. Using Inertial Measurement Units to Examine Selected Joint Kinematics in a Road Cycling Sprint: A Series of Single Cases. *Sensors* **2024**, *24*, 3453. https://doi.org/10.3390/s24113453

Academic Editors: Christian Peham, Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 26 February 2024
Revised: 6 May 2024
Accepted: 23 May 2024
Published: 27 May 2024

1. Introduction

In competitive cycling, sprinting is an essential component that determines the result of many races. In the three Men's Grand Tours, The Giro d'Italia, Tour de France, and Vuelta a España, one in every three stages are decided by a mass, small bunch, or a head-to-head sprint [1–3]. A sprint is defined by a sudden increase in power output and effort leading to a sustained acceleration [1]. This can happen in the closing meters of the race as competitors attempt to be the first cyclist across the finish line [2]. In this situation, the sprints are most commonly completed in an out-of-saddle (standing) position; however, for longer sprints or sustained attacks during a particular stage, these sprint-type efforts may be completed in an in saddle (seated) position. Menaspà and colleagues [2] determined that sprints in the men's races last an average of 13 s in duration, with an average speed of 64 km/h, while Peiffer et al. [4] reported that sprints in women's races typically last 22 s, with an average speed of 54 km/h. Despite being such an important component in competitive cycling, there is a paucity of information concerning the kinematics involved in the action of sprinting.

Most research investigating biomechanics in road cycling has been performed in a controlled laboratory setting and does not focus on the action of sprinting. For example, Bertucci et al. [5] compared the difference in crank torque and the rate of perceived exertion (RPE) of riding on an ergometer in a lab against riding a bicycle outdoors. They observed notable differences in both variables, with increases recorded for values collected on the ergometer. This is believed to be due to an increased stiffness and damping of forces on the ergometer, as well as the fact that the rider is overcoming the load of the flywheel on the ergometer opposed to their own mass on the bicycle outdoors [6]. Changes in environment (laboratory vs. outdoors) are likely to influence the cyclist's biomechanics during sprint cycling, particularly through the upper limbs, as the front steering mechanism of a bicycle is not fixed, unlike an ergometer [6]. Further, a systematic review by Johnston et al. [7] investigating cycling knee biomechanics reported that all 14 studies included were completed in a controlled laboratory on either a stationary ergometer or stationary bicycle. These researchers drew similar conclusions to Fregly et al. [6] and Burnie et al. [8], who highlighted the limitations in knowledge when a biomechanical analysis is not conducted in an environment representative of that in which the sport is competed [7].

Previously, Costes et al. [9] reported that the upper limb transfers 3–5% of energy into total crank power output during cycling. This upper limb function becomes more critical with high-intensity cycling, in which the power output is reduced to 10–20% if upward forces are not produced on the handlebar [10]. During seated cycling, the energy produced by muscles is delivered into three points: pedals, seat, and handlebars [11]. It has been hypothesized the upper limb joints stabilize the trunk as well as creating seat force and transferring energy down to the lower limb. An increase in the force produced by the legs leads to an increase in the upward force at the hip joint. This force production results in a decrease in the seat reaction force, which requires cyclists to compensate by generating upper-limb force to pull on the handlebar until they can reach an adoption point [12]. Moreover, alternate bilateral movements of the left and right leg result in decreased trunk stabilization, which results in the upper limb absorbing extra force to increase stability during cycling [9]. These findings are derived from an understanding of the mechanical movements in cycling. However, no empirical kinematic data support these hypotheses. Thus, this research aimed to describe selected joint kinematics during a cycling sprint outdoors. We sought to evaluate two different types of cycling sprints: standing and seated sprinting, comparing these to a seated baseline condition of cycling at 20 km/h. We also piloted a running gait analysis approach to process the cycling kinematics, allowing insights from multiple pedal revolutions to be assessed.

2. Materials and Methods

We recruited three participants for this cross-section, case series study design. Each participant was judged to be "Tier 3" or "Highly Trained/National Level" based on the 6-tier system proposed by McKay et al. [13]. Participant 1, who is female (age: 21 years, weight: 56 kg, height: 1.55 m), competes at an elite and U23 level in international triathlon competition and is ranked in the Top 500 in the World Triathlon Rankings and the Top 70 in the Asian Triathlon Rankings. Participant 2, who is male (age: 20 years, weight: 69 kg, height: 1.79 m), competes at a U23 level in international triathlon competition and is ranked in the Top 15 of the U23 Triathlon Australia Rankings. Participant 3, who is male (age: 19 years, weight: 76 kg, height: 1.87 m) competes at a U23 level in international triathlon competition and is ranked in the Top 15 of the U23 Triathlon Australia Rankings. Although these participants were from a triathlon background, they had predominantly competed in draft-legal races more akin to cycling criterium racing. They were also coached by a nationally accredited cycling coach. All participants provided written consent prior to any data collection and the study was approved by the Curtin University Human Research Ethics Committee (HRE2019-0418).

Data collection occurred over a one-week period in August and September 2023 at an outdoor cycling track in Manning, Western Australia (Temperature: 20.6–29.8 °C, Humidity: 31.5–57.5%). The cycling track was an 800 m long loop including a 150 m straight section. The entire track was used for the 10 min warm-up but only the straight portion was to collect sprinting data. During the sprints, three-dimensional trunk, upper and lower limb kinematics were measured at 200 Hz via a set of wireless inertial measurement units (IMU) (Noraxon Myomotion, Scottsdale, AZ, USA) using previously reported placements [14,15].

Briefly, 15 sensors were attached to participants over the following locations: C7, T12, L5, pelvis and bilaterally on the hands, forearms, upper arms, thighs, shanks, and feet. Each sensor has an accelerometer of ±200 g, a gyroscope of ±7000°/s and a magnetometer of 16 gauss. The system was previously reported to have a valid evaluation of kinematics in team sports compared to traditional optical motion capture [15].

Upon arrival at the track, participants' body mass (while wearing their standard cycling attire) and standing height were measured (using a Transtek BS-801-BT Body Scale and Craftright 30 m Tape Measure, respectively). Next, participants had IMUs attached to them, according to the manufacturer's specifications, in the positions described. In addition to the IMU, a single global positioning system (GPS) (Catapult S5, Catapult Sports, Melbourne, Australia) collecting samples at 10 Hz was attached to each participant's bicycle seat post to measure their velocity profile. All participants used their own bike, shoes, and cycling clothing so as not to interfere with the personalized set-ups cyclists normally adopt. We understand that allowing participants to use their own equipment introduces potential biases to the data; however, we endeavored to follow an ecologically valid approach, as all previous studies had only tested cyclists in a laboratory, merely making inferences to real-world cycling challenging. Following this, participants completed a 10 min warm-up at a self-selected effort with three 6 s familiarization sprints at minutes 6, 7, and 8 post-start [8,16].

After the warm-up, data collection commenced. First, participants were asked to simply cycle seated through the outdoor course at a constant speed of 20 km/h. This provided a baseline condition from which to compare kinematic changes during seated and standing sprinting. Participants completed the baseline condition once. Then, participants were recorded while sprinting within a 60 m "testing zone" which was set and marked on the 150 m straight section of the course. Prior to commencing the sprint, participants entered the testing zone at a speed of approximately 20 km/h, using their bicycle computer to monitor their speed. This procedure was completed six times, with the first three sprints completed standing (out of saddle) and the last three sprints seated (in saddle) [3]. After each sprint effort, participants were provided with a 3 min recovery period.

The GPS data were uploaded to manufacturer-supplied software (Catapult Sports, Melbourne, Australia). For each of the participants' six trials, the maximum velocity achieved was determined for both in- and out-of-saddle conditions. Only the best trial (out of the three available) based on the maximum velocity as recorded by the GPS was analyzed further for kinematics. To determine this, three-dimensional kinematics were extracted from the IMUs using the manufacturer-supplied software (Noraxon myoResearch ver3.18, Scottsdale, AZ, USA). The kinematic data was sectioned into single pedal cycles for the middle eight revolutions of the 60 m seated, standing, and control conditions. An automatic algorithm detected the minimum knee flexion of the first revolution to the next minimum knee flexion of the next revolution. This was performed for both left and right knees and delineated all the kinematics into separate single cycles. Specific kinematics were selected and analyzed. This selection was based on previous work which emphasized the importance of the upper limb and trunk in sprinting [12] and included bilateral wrist flexion/extension, elbow flexion/extension, shoulder flexion/extension, and abduction/adduction. The truck kinematics analyzed included thoracic and lumbar spine flexion/extension, lateral flexion and axial rotation. Further, bilateral lower limb kinematics, including hip flexion/extension and abduction/adduction, knee flexion/extension, and ankle plantar/dorsiflexion, were also examined. Each cycle of the selected kinematics was then time-normalized to 0–100% of a cycle. The eight time-normalized revolutions for the selected kinematics were averaged to obtain an average (±SD) kinematic curve similar to those produced during typical running gait biomechanical analysis. Lastly, acceleration values for each participant were calculated using data recorded by the GPS.

The average, minimum and maximum joint excursion for the selected kinematics were obtained for the trunk and bilaterally for the upper and lower limbs. As this study was a series of three cases examining methods to obtain and process in-field cycling kinematic data, only descriptive statistics (mean and standard deviations (±SD)) were reported.

3. Results

A lower peak velocity was observed during the in-saddle (Figure 1) conditions compared to out-of-saddle conditions (Figure 2). This was consistent across each of the participants. Participant 1 reached a peak velocity of 11.6 m/s at an acceleration of 0.7 m/s^2 for standing sprinting and 10.4 m/s at an acceleration of 0.5 m/s^2 for seated sprinting. Participant 2 reached 13.1 m/s at an acceleration of 0.8 m/s^2 for standing sprinting and 11.7 m/s at an acceleration of 0.6 m/s^2 for seated sprinting, while Participant 3 reached 13.9 m/s at an acceleration of 0.9 m/s^2 during standing sprinting and 13.0 m/s at an acceleration of 0.9 m/s^2 during seated sprinting.

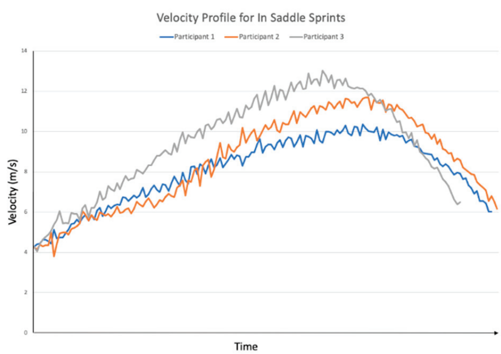


Figure 1. Velocity over time for each participant during the seated sprint condition.

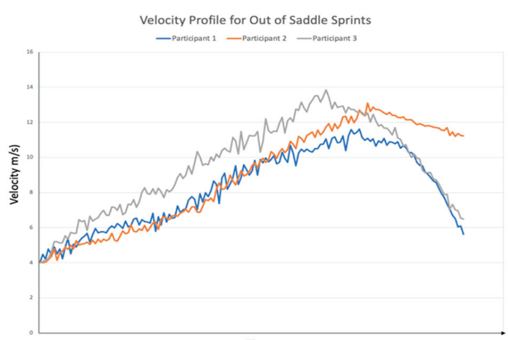
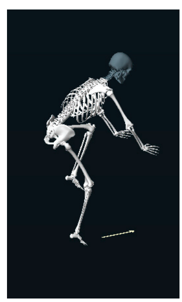
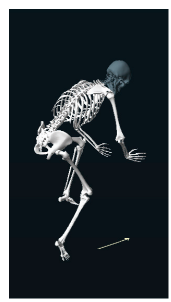
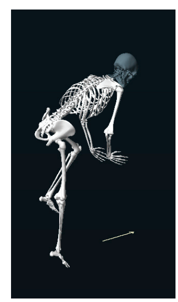
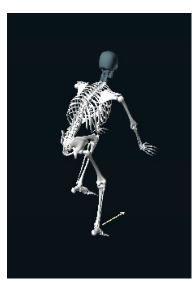
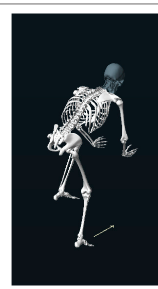
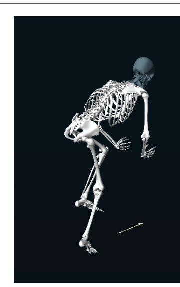
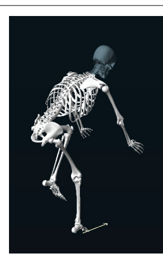
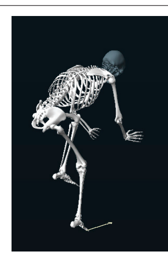
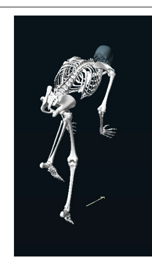


Figure 2. Velocity over time for each participant during the standing sprint condition.

Upper limb kinematic analysis revealed that participants maintained a stable joint angle through the pedal stoke during baseline cycling, with very small changes (≈5°) in their range of movement (Table 1 and Figure 3A,D,G,J,M). During seated sprinting, wave-like patterns started to emerge with increases in joint excursions (≈20°). When participants performed standing sprinting, discernable cyclic patterns emerged, especially for wrist flexion and extension (Figure 3C) and shoulder abduction and adduction (Figure 3O). Other joint kinematics did seem to display a wave-like pattern, but individual differences also emerged. For example, Participant 2's elbow and shoulder flexion and extension displayed a different pattern to that of Participant 1 and 3 (Figure 3I,L). These differences occurred during the up phase of the pedal stroke for the elbow and throughout the whole stroke for the shoulder. Larger joint excursion (30–40°) was also noted in wrist and elbow flexion and extension and shoulder abduction and adduction. Shoulder flexion reduced by approximately 20° during standing sprinting compared to seated sprinting and baseline cycling.

In general, participants adopted a flexed lumbar spine and an extended thoracic spine during baseline cycling (Table 1 and Figure 4). Further lateral flexion and axial rotation in the lumbar and thoracic spine were close to neutral during baseline cycling; however, there were individual differences (Figure 4A,D,G,J,M,P). Participant 2 adopted a posture of left laterally flexed thoracic spine and a right laterally flexed lumbar spine during baseline cycling (Figure 4D,M). Participants adopted approximately 5°–10° less lumbar flexion as they moved from baseline cycling through to standing sprinting (Figure 4J–L). However, Participants 1 and 3 adopted a more neutral thoracic spine posture while Participant 2 maintained an extended thoracic spine while sprinting in both seated and standing positions (Figure 4B,C). Small joint excursions (approximately 10° peak-to-peak) were recoded for thoracic and lumbar lateral flexion and axial rotation movements (Figure 4F,I). Thoracic axial rotation also displayed a wave-like pattern during the standing sprint (Figure 4I).

Table 1. Avatars of typical postures adopted by participants for each condition. Near-top dead center (left leg)/bottom dead center (right leg) are depicted. Please note that as the head position was not measured, the software has assumed a neutral head position was maintained relative to the spinal column and grayed the head and cervical spine.

	Baseline	**Seated**	**Standing**
Participant 1			
Participant 2			
Participant 3			

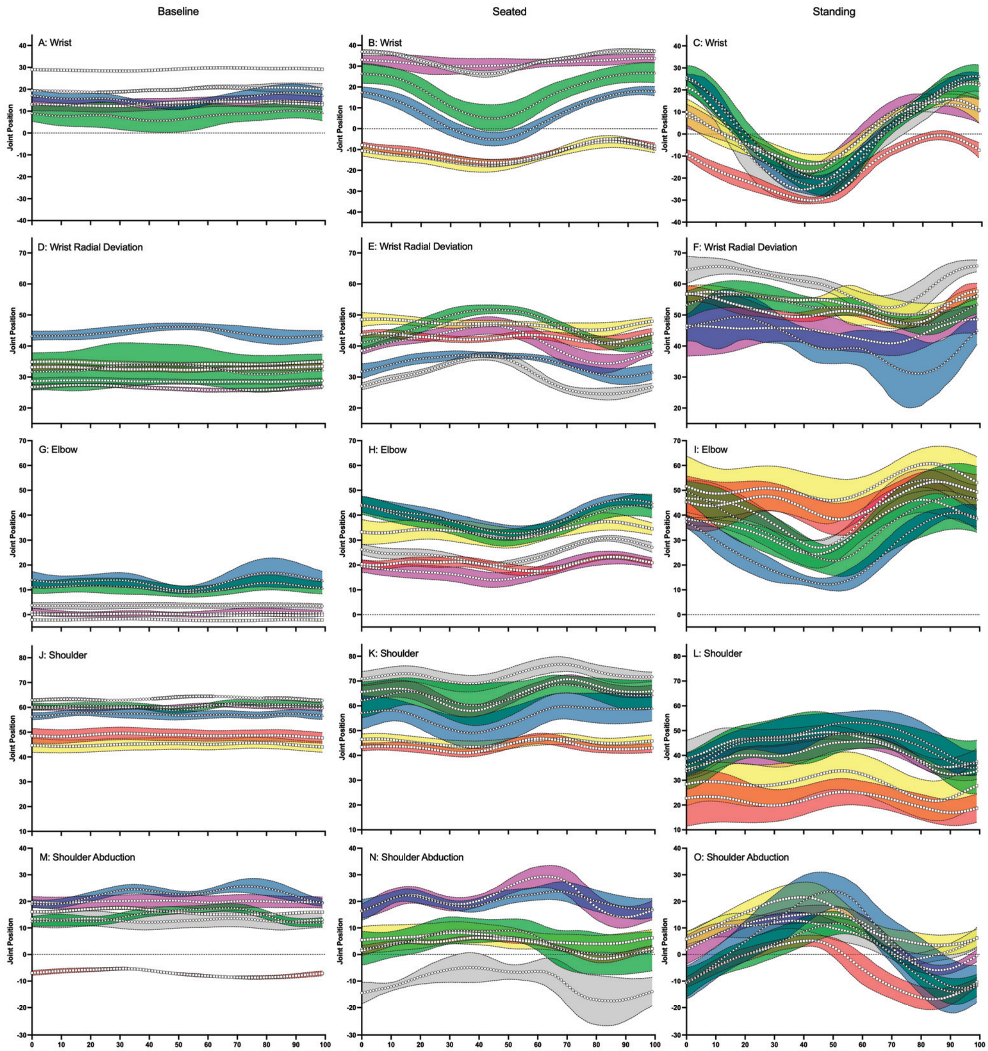


Figure 3. This figure illustrates the wrist flexion (positive) and extension (negative) (**A**–**C**), wrist radial (positive) and ulnar (negative) deviation (**D**–**F**), elbow flexion (positive) and extension (negative) (**G**–**I**), shoulder flexion (positive) and extension (negative) (**J**–**L**) and shoulder abduction (positive) and adduction (negative) (**M**–**O**) average joint positions and standard deviation across baseline, seated and standing cycling conditions for Participant 1 (△: blue—left and green—right), Participant 2 (□: red—left and yellow—right), Participant 3 (○: gray—left and purple—right). In each panel where applicable the zero (0) has been shown with a dotted line.

Hip and knee flexion and extension followed a very similar pattern for all participants in the three conditions tested (Table 1 and Figure 5A–C,G–I). Greater hip and knee extension (up to 10°) were recorded in all participants during the standing sprints (Table 1 and Figure 5C,I). Hip abduction and adduction and ankle dorsiflexion and plantar flexion were varied during baseline cycling (Figure 5D,J). These kinematics developed into more discernable, wave-like patterns when participants performed the seated and standing sprints (Figure 5E,F,K,L).

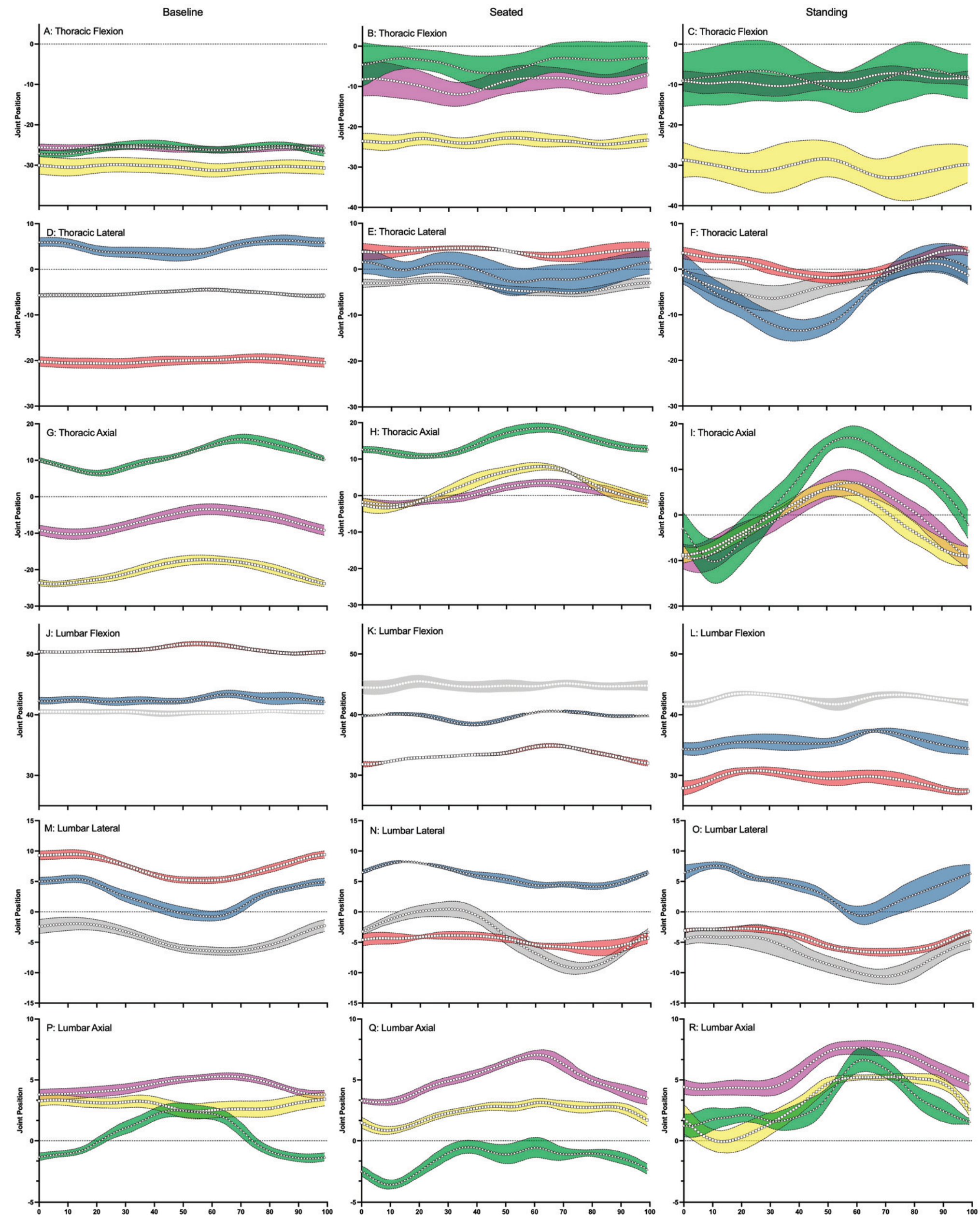


Figure 4. This figure illustrates the thoracic flexion (positive) and extension (negative) (**A**–**C**), thoracic lateral flexion (positive = right lateral flexion, negative = left lateral flexion) (**D**–**F**), thoracic axial rotation (positive = right axial rotation, negative = left axial rotation) (**G**–**I**), lumbar flexion (positive) and extension (negative) (**J**–**L**) lumbar lateral flexion (positive = right lateral flexion, negative = left lateral flexion) (**M**–**O**), lumbar axial rotation (positive = right axial rotation, negative = left axial rotation) (**P**–**R**) positions and standard deviation across baseline, seated, and standing cycling conditions for Participant 1 (△: blue—lateral and green—axial), Participant 2 (□: red—lateral and yellow—axial), Participant 3 (○: gray—lateral and purple—axial). In each panel where applicable the zero (0) has been shown with a dotted line.

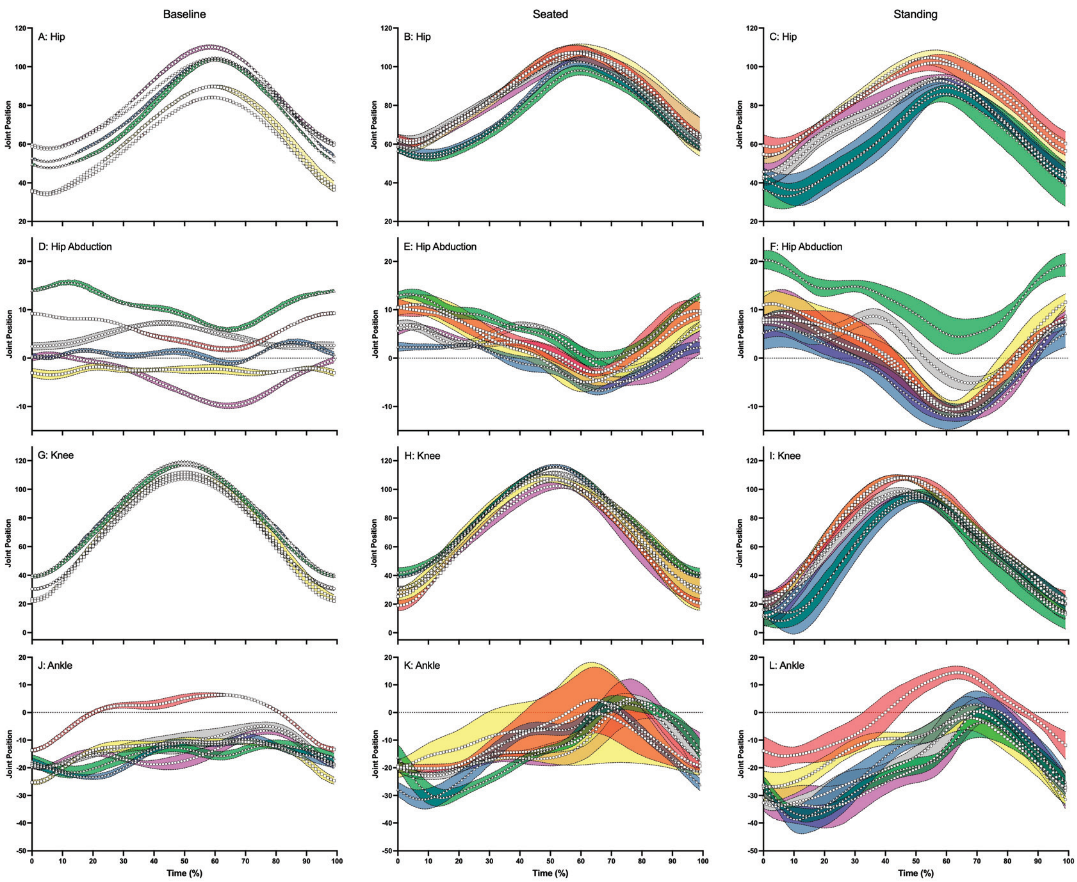


Figure 5. Illustrates the hip flexion (positive) and extension (negative) (**A**–**C**), hip abduction (positive) and adduction (negative) (**D**–**F**), knee flexion (positive) and extension (negative) (**G**–**I**) and ankle dorsiflexion (positive) and plantarflexion (negative) (**J**–**L**) average joint positions and standard deviation across baseline, seated and standing cycling conditions for Participant 1 (△: blue—left and green—right), Participant 2 (□: red—left and yellow—right), Participant 3 (○: gray—left and purple—right). In each panel where applicable the zero (0) has been shown with a dotted line.

4. Discussion

This study aimed to describe selected kinematics during road cycle sprints and to understand the influence that completing the sprint seated versus standing had on these kinematics. Further, a method of analyzing cycling kinematics similarly to running kinematics was trialed to judge the utility of this approach. We found that standing sprinting produced greater final velocity and acceleration compared to seated sprinting. The upper limb and trunk kinematics remained stable during baseline cycling but discernable patterns started to emerge during seated sprinting and were much more pronounced and obvious during standing sprinting. Lower limb kinematics tended to follow distinct patterns, especially sagittal plane movement in the hips and knees. Ankle moment showed more variation during baseline cycling but was more pattern-like during sprinting.

The velocity profiles observed from out-of-saddle and in-saddle sprinting differed in this study. There was a difference in final velocity for standing versus seated sprinting, which was consistent for all participants. It has been shown that greater power output occurs during standing sprint-cycling versus seated sprint-cycling [17,18], and this can explain the difference noted in the present study. Further, a previous biomechanical review described the relationship between increased leg-produced force, increased hip-reaction force, and decreased seat-reaction force during seated sprints and suggested cyclist compensate for this by pulling on the handlebars until they stand and sprint out of saddle [11]. Stone and Hull [19] suggested during standing sprinting, the hips are placed further forward compared to seated sprinting, creating greater crank arm-leverage, which explains the increased speed observed during this form of sprinting. The shoulder flexion kinematics from our study support this hypothesis, as decreased shoulder flexion was observed in all three participants during the standing sprint. Further investigations involving greater numbers of cyclists of all abilities are needed to confirm these findings.

Analysis of upper body kinematics showed that participants adopted a more stable posture during baseline cycling, with increasing ranges of movement as they sprinted. Holliday et al. [20] also showed larger average movement in the elbow during more intense (90% of max) compared to less intense cycling (60% of max). The average elbow flexion values reported by Holliday et al. [20] are similar to the values we obtained from participants during seated sprinting compared to higher levels of cycling intensity. These authors also reported little change in the shoulder flexion angle between the intensities tested, which is similar to our observations when our baseline condition was contrasted against seated sprinting. It was only during the standing sprint when shoulder flexion angles decreased in all participants. This can be explained by Stone and Hull's [19] hypothesis that cyclists place their hips further forward during standing sprints. Only small shoulder abduction and adduction ranges were recorded in our study and there was variance in this angle between participants, especially during the baseline and seated sprint conditions. This can be attributed to the differences in handlebar width between our participants' bicycles, especially considering that the difference in ranges and postures diminished during the standing sprint, where handlebar width may be less of a factor. Shoulder abduction and adduction movement may be an important consideration for bicycle fit and future studies may want to consider this movement in their research.

Concerning wrist joint excursions, although there were smaller ranges of movement in wrist flexion and extension as well as wrist radial and ulnar deviation during baseline cycling, these ranges increased during seated and standing sprinting. Further, differences between participants and differences between left and right wrists were also noted, especially for wrist radial and ulnar deviation. In fact, some of the values obtained for wrist radial deviation were potentially close to a full range of motion [21]. These findings are of interest, considering that radial deviation may be of more significance for cycling performance than previously considered. The wrist joint may play a similar role in sprint cycling performance to that which the ankle joint plays sprint running performance. Martín-Fuentes & van den Tillaar [22] found that the time from dorsal flexion to toe-off had a significant impact on performance among sprint runners. Future research describing the relationship between sprint cycling performance and time and radial deviation range should seek to determine if the wrist in sprint cycling is like the ankle in sprint running. If such an association is found, specific training can be considered for the improvement of sprint cycling performance, as ankle-specific training is already integrated in training for elite sprint running [23]. Wrist kinematics should be further investigated in a larger sample of track cyclists to elucidate the role the wrist plays in cycling sprinting.

The trunk kinematics recorded in our study showed that participants adopted a forward-flexed posture in the lumbar spine with smaller ranges in the frontal and transverse plane for both the lumbar and thoracic spine. Participants' forward flexed posture in the lumbar spine decreased during sprinting. Participants also adopted an extended posture in the thoracic spine, and this extension also decreased as participants sprinted. Our lumbar spine results are similar to those previously reported, but our thoracic spine results differ substantially from those of other researchers [20]. The differences can be attributed to two main factors. Firstly, Holliday et al. [20] calculated and reported thoracic spine orientation angle relative to the laboratory coordinate system while our thoracic spine is reported relative to the lumbar spine. Our method has been shown to be a valid approach to understanding spinal kinematics in fast bowling during cricket games [24]. Secondly, Holliday et al. [20] tested participants using an ergometer in a laboratory, while our data were obtained during field-based cycling. It may be that during cycling, it is paramount to have an extended thorax for forward gaze to allow cyclist to gauge road conditions, maintain balance, and avoid obstacles. Frontal and transverse plane kinematics were smaller in magnitude and range compared to those recorded in the sagittal plane. However, substantial differences were recorded between participants and cycling conditions. For example, Participant 2 adopted a somewhat scoliotic posture during baseline cycling with pronounced left lateral flexion in the thoracic spine offset by right lateral flexion in the lumbar spine. This posture did not manifest during seated or standing sprinting, in which a more neutral spine posture was adopted. More research is needed to further understand trunk kinematics in cycling.

The lower limb kinematics in our study demonstrate a distinct pattern of movement, especially in the hips and knees in the sagittal plane during all conditions, the hips in the frontal plane, and the ankle in the sagittal plane during the sprinting conditions. Our results for hip flexion and extension, hip abduction and adduction, and knee flexion and extension during baseline cycling show similar patterns to those reported by Yum et al. [25]. However, the magnitudes of the ranges of motion are larger in our study. This may be attributed to the fact our participants rode at 20 km/h during baseline cycling while Yum et al. [25] had their participants ride at 10–12 km/h on an ergometer. In contrast, our results recorded during seated sprinting for knee extension at bottom dead center are similar in magnitude to those found in the work of Holliday et al., [20]. However, our results for hip extension at top dead-center are lower compared to these authors' results. Kinematics derived from IMUs have been shown to be very similar to optical motion capture data, especially in the knee, but variances have been recorded for hip kinematics, and this may, in part, explain the discrepancy in results.

Our results for ankle dorsi and plantar flexion during baseline cycling differed from those of Yum et al. [25], with our participants adopting a more plantarflexed posture. Our participants used clipless pedals to attach their shoes to the bicycle, while pictures of Yum et al.'s [25] configuration suggested their participants rode barefoot on an ergometer. The use of clipless pedals allows cyclists to exert pull forces during the upward phase of the pedal stroke and, as such, allows them to exhibit a more plantarflexed foot. Our results also show that participants adopted more knee extension and ankle dorsiflexion range as they sprinted. These results are in agreement with the results of others who have also showed these changes with incremental cycling intensities. However, our results show little change in the hip extension range, while others have shown increases. Our participants needed to adopt postures that allowed forward vision, while the other researchers cited performed their experiments in a laboratory where forward vision was not prioritized.

A further confounder of comparison with the available literature is the potential differences between ergometer frame stiffness and actual bike frame stiffness characteristics and how these can influence cyclists' kinematics. During seated cycling, the energy produced by muscles is delivered to three points: pedals, seat, and handlebars [11]. Baker and Davies [10] reported the importance of upper limb function during high-intensity cycling and the role of the handlebar; this was conducted on a relatively stable laboratory ergometer. Building on this finding, Costes et al. [9] highlighted the upper limb energy-transfer contributions (3–5%) to total crank power output during real-world cycling, but found that this is compromised if upward forces are not produced on the handlebar. Turpin et al. [12] hypothesized how upper limb joints stabilize the trunk to create seat force and transfer energy down to the lower limb. While our current data do not provide insight on how the forces produced by the legs interact with seat reaction force, we do provide initial evidence on the truck kinematics and how these seek to compensate for system stiffness differences between seated and standing sprinting and the handlebar interaction.

Our method of collecting, analyzing, and reporting cycling kinematics has shown that field-based cycling kinematics do vary from those observed during laboratory-based studies and that extrapolating results from the laboratory may be problematic. Further, the characterization of whole pedal stroke data, like the approach used in running gait analysis, shows promise, as these data are rich and insightful. Features such as movement variability as well as bilateral differences during the whole pedal stroke can be analyzed. Further, subtle differences between cyclists can also be studied. This may improve our understanding of this popular sport and may lead to performance enhancements, injury prevention, and optimal rehabilitation and return to cycling after injury. It may also give us insights into errors made by cyclists, which can result in serious crashes and injury.

Our study is limited by the small sample size and, thus, the ability to extrapolate the findings to a broader population. Consistent findings across a larger sample of cyclists are needed to improve the level of evidence. Furthermore, our participants are triathletes, and although they were familiar with criterium racing as well as being coached by a nationally accredited cycling coach, potential inclusion of varying levels of road cyclists and/or track cyclists is needed. This approach will ensure a rich data source for the optimization of cycling. Allowing participants to use their own equipment can easily add to the variance obtained in our kinematics. Further, environmental factors such as wind cannot be controlled or accounted for. Our approach, however, allows scientists interested in cycling to study these phenomena, allowing us to expand our knowledge about cycling.

5. Conclusions

Our study showed that our participants maintained a stable posture in most joints studied during baseline cycling, but substantial changes in kinematics were noted as they performed seated and standing sprints. In particular, discernable patterns started to emerge in the upper limb joints and the ankle. Specific postures in the trunk were maintained depending on the cycling activity. Our approach also showed that although the patterns of kinematics in many joints were similar to those reported in previous laboratory studies, the magnitude of the ranges of movement do differ. Our approach also highlighted insightful results where movement variability within and between cyclists can be studied. Future research should examine the in-field kinematics of a larger sample of road and track cyclists during sprinting. This could facilitate a definitive examination of the association of certain kinematic strategies with a superior sprinting performance. This information will greatly enhance coaching and training strategies in competitive cycling.

Author Contributions: Conceptualization, D.W.C. and K.N.; methodology, S.M., M.T., K.K., D.W.C. and K.N.; formal analysis, S.M., M.T., K.K., D.W.C. and K.N.; writing—S.M., M.T. and K.K.; writing—review and editing, D.W.C. and K.N.; visualization, D.W.C. and K.N.; supervision, D.W.C. and K.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of Curtin University (HRE2019-0418, 1 August 2023).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Raw kinematic data from this study can be obtained by contacting the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Menaspà, P.; Abbiss, C.R.; Martin, D.T. Performance analysis of a world-class sprinter during cycling Grand Tours. *Int. J. Sports Physiol. Perform.* **2013**, *8*, 336–340. [CrossRef] [PubMed]
2. Menaspà, P.; Quod, M.; Martin, D.T.; Peiffer, J.J.; Abbiss, C.R. Physical Demands of Sprinting in Professional Road Cycling. *Int. J. Sports Med.* **2015**, *36*, 1058–1062. [CrossRef] [PubMed]
3. Merkes, P.; Mensapà, P.; Abbiss, C. Sprinting in road cycling—Literature review. *J. Sci. Cycl.* **2020**, *9*, 29–52. [CrossRef]
4. Peiffer, J.J.; Abbiss, C.R.; Haakonssen, E.C.; Menaspà, P. Sprinting for the Win: Distribution of Power Output in Women's Professional Cycling. *Int. J. Sports Physiol. Perform.* **2018**, *13*, 1237–1242. [CrossRef] [PubMed]
5. Bertucci, W.; Grappe, F.; Alain, G. Laboratory versus Outdoor Cycling Conditions: Differences in Pedaling Biomechanics. *J. Appl. Biomech.* **2007**, *23*, 87–92. [CrossRef] [PubMed]
6. Fregly, B.J.; Zajac, F.E.; Dairaghi, C.A. Bicycle drive system dynamics: Theory and experimental validation. *J. Biomech. Eng.* **2000**, *122*, 446–452. [CrossRef] [PubMed]
7. Johnston, T.E.; Baskins, T.A.; Koppel, R.V.; Oliver, S.A.; Stieber, D.J.; Hoglund, L.T. The influence of extrinsic factors on knee biomechanics during cycling: A systematic review of the literature. *Int. J. Sports Phys. Ther.* **2017**, *12 pt 20171023*, 1023–1033. [CrossRef] [PubMed]
8. Burnie, L.; Barratt, P.; Davids, K.; Worsfold, P.; Wheat, J.S. Effects of strength training on the biomechanics and coordination of short-term maximal cycling. *J. Sports Sci.* **2022**, *40*, 1315–1324. [CrossRef] [PubMed]
9. Costes, A.; Turpin, N.; Villeger, D.; Moretto, P.; Watier, B. Influence of Position and Power Output on Upper Limb Kinetics in Cycling. *J. Appl. Biomech.* **2016**, *32*, 140–149. [CrossRef]
10. Baker, J.S.; Davies, B. Additional Considerations and Recommendations for the Quantification of Hand-Grip Strength in the Measurement of Leg Power During High-Intensity Cycle Ergometry. *Res. Sports Med.* **2009**, *17*, 145–155. [CrossRef]
11. Turpin, N.A.; Watier, B. Cycling Biomechanics and Its Relationship to Performance. *Appl. Sci.* **2020**, *10*, 4112. [CrossRef]
12. Turpin, N.; Costes, A.; Moretto, P.; Watier, B. Upper limb and trunk muscle activity patterns during seated and standing cycling. *J. Sports Sci.* **2016**, *35*, 557–564. [CrossRef] [PubMed]
13. McKay AK, A.; Stellingwerff, T.; Smith, E.S.; Martin, D.T.; Mujika, I.; Goosey-Tolfrey, V.L.; Sheppard, J.; Burke, L.M. Defining Training and Performance Caliber: A Participant Classification Framework. *Int. J. Sports Physiol. Perform.* **2022**, *17*, 317–331. [CrossRef] [PubMed]
14. Camomilla, V.; Bergamini, E.; Fantozzi, S.; Vannozzi, G. Trends Supporting the In-Field Use of Wearable Inertial Sensors for Sport Performance Evaluation: A Systematic Review. *Sensors* **2018**, *18*, 873. [CrossRef] [PubMed]
15. Heuvelmans, P.; Benjaminse, A.; Bolt, R.; Baumeister, J.; Otten, E.; Gokeler, A. Concurrent validation of the Noraxon MyoMotion wearable inertial sensors in change-of-direction and jump-landing tasks. *Sports Biomech.* **2022**, 1–16. [CrossRef] [PubMed]
16. Minahan, C.; Sheehan, B.; Doutreband, R.; Kirkwood, T.; Reeves, D.; Cross, T. Repeated-sprint cycling does not induce respiratory muscle fatigue in active adults: Measurements from the powerbreathe® inspiratory muscle trainer. *J. Sports Sci. Med.* **2015**, *14*, 233–238. [PubMed]
17. Merkes, P.F.J.; Menaspà, P.; Abbiss, C.R. Power output, cadence, and torque are similar between the forward standing and traditional sprint cycling positions. *Scand. J. Med. Sci. Sports* **2020**, *30*, 64–73. [CrossRef] [PubMed]
18. Samozino, P.; Peyrot, N.; Edouard, P.; Nagahara, R.; Jimenez-Reyes, P.; Vanwanseele, B.; Morin, J.B. Optimal mechanical force-velocity profile for sprint acceleration performance. *Scand. J. Med. Sci. Sports* **2022**, *32*, 559–575. [CrossRef] [PubMed]
19. Stone, C.; Hull, M.L. Rider/Bicycle Interaction Loads during Standing Treadmill Cycling. *J. Appl. Biomech.* **1993**, *9*, 202–218. [CrossRef]
20. Holliday, W.; Theo, R.; Fisher, J.; Swart, J. Cycling: Joint kinematics and muscle activity during differing intensities. *Sports Biomech.* **2023**, *22*, 660–674. [CrossRef]
21. Li, Z.-M.; Kuxhaus, L.; Fisk, J.A.; Christophel, T.H. Coupling between wrist flexion–extension and radial–ulnar deviation. *Clin. Biomech.* **2005**, *20*, 177–183. [CrossRef] [PubMed]
22. Martín-Fuentes, I.; van den Tillaar, R. Relationship between Step-by-Step Foot Kinematics and Sprint Performance. *Int. J. Environ. Res. Public Health* **2022**, *19*, 6786. [CrossRef] [PubMed]
23. Bezodis, I.N.; Kerwin, D.G.; Cooper, S.M.; Salo AI, T. Sprint Running Performance and Technique Changes in Athletes During Periodized Training: An Elite Training Group Case Study. *Int. J. Sports Physiol. Perform.* **2018**, *13*, 755–762. [CrossRef] [PubMed]
24. Harnett, K.; Plint, B.; Chan, K.Y.; Clark, B.; Netto, K.; Davey, P.; Müller, S.; Rosalie, S. Validating an inertial measurement unit for cricket fast bowling: A first step in assessing the feasibility of diagnosing back injury risk in cricket fast bowlers during a tele sport-and-exercise medicine consultation. *PeerJ* **2022**, *10*, e13228. [CrossRef]
25. Yum, H.; Kim, H.; Lee, T.; Park, M.S.; Lee, S.Y. Cycling kinematics in healthy adults for musculoskeletal rehabilitation guidance. *BMC Musculoskelet. Disord.* **2021**, *22*, 1044. [CrossRef]

sensors

Article

Playing Flappy Bird Based on Motion Recognition Using a Transformer Model and LIDAR Sensor

Iveta Dirgová Luptáková, Martin Kubovčík * and Jiří Pospíchal *

Institute of Computer Technologies and Informatics, Faculty of Natural Sciences, University of Ss. Cyril and Methodius, J. Herdu 2, 917 01 Trnava, Slovakia; iveta.dirgova.luptakova@ucm.sk
* Correspondence: martin.kubovcik@ucm.sk (M.K.); jiri.pospichal@ucm.sk (J.P.)

Abstract: A transformer neural network is employed in the present study to predict Q-values in a simulated environment using reinforcement learning techniques. The goal is to teach an agent to navigate and excel in the Flappy Bird game, which became a popular model for control in machine learning approaches. Unlike most top existing approaches that use the game's rendered image as input, our main contribution lies in using sensory input from LIDAR, which is represented by the ray casting method. Specifically, we focus on understanding the temporal context of measurements from a ray casting perspective and optimizing potentially risky behavior by considering the degree of the approach to objects identified as obstacles. The agent learned to use the measurements from ray casting to avoid collisions with obstacles. Our model substantially outperforms related approaches. Going forward, we aim to apply this approach in real-world scenarios.

Keywords: reinforcement learning; motion sensors; ray casting; signal processing; time series processing; transformer model; robotics; Flappy Bird game; agent control

Citation: Dirgová Luptáková, I.; Kubovčík, M.; Pospíchal, J. Playing Flappy Bird Based on Motion Recognition Using a Transformer Model and LIDAR Sensor. *Sensors* **2024**, *24*, 1905. https://doi.org/10.3390/s24061905

Academic Editors: Arturo de la Escalera Hueso, Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 22 December 2023
Revised: 7 March 2024
Accepted: 14 March 2024
Published: 16 March 2024

1. Introduction

The autonomous control of robots using reinforcement learning (RL) has emerged as one of the important topics in machine learning. The extensive use of deep neural network technology has made it the most common choice for creating control systems that rely on information collected from a robot's operating environment. This paper focuses on processing the collected data within a time framework and using motion information to control the robot's actions. The architecture used is the transformer model [1], which can efficiently process long time series [2].

The popular computer game Flappy Bird created by Vietnamese programmer Dong Nguyen [3] acts as the simulation environment here. The goal of the player, who controls a simulated robot bird, is to fly continuously forward without a collision. The bird encounters a succession of pairs of pipes obstructing its path, and they are suspended from the top and protrude from the bottom of the game environment. A constant distance is maintained between each pair of pipes, forming a gap through which the bird can fly. The vertical position of this gap is randomly generated, introducing a dynamic element to the game. The ever-changing environment demands that players adapt and quickly react. Gravity pulls the bird downward, whereas the player's actions push the bird upward. Horizontal velocity remains constant. The game concludes instantly if the bird collides with either a pipe or the ground.

There are several approaches to train players in Flappy Bird. One typical approach is to use the image generated by the game [4], with various adaptations. Another modification involves introducing an extra negative reward when the agent collides with the upper edge of the game screen [5]. Further modifications are based on the creation of three training difficulty levels, easy, medium, and hard [6], which are distinguished by the width of the gap between pipes. The subsequent method involved a computer expert who extracted key information from the pipes and the agent, which is then used to predict actions [7].

In this paper, the player–bird is equipped with a simulated light detection and ranging (LIDAR) sensor represented by the ray casting method to detect pipelines and ground. The player can utilize time-series signal processing to maneuver around pipelines and avoid collisions. The goal is to use motion data to navigate through obstacles, such as pipes and the ground. In this model, a custom-built deep neural network, called here the "motion transformer", is employed for both time series and ray casting signal processing.

A similar approach to processing temporal components is used by [8–10]. However, these papers primarily focus on processing human activity data while also incorporating spatial components. All spatial measurements are interpreted as features. The transformer model in the present study is only looking for time correlations and not for correlations between the rays of the sensor.

The transformer model has already been used for action prediction, where it determines the next action based on the current state, previous actions, and rewards. However, this model specifically uses a causal transformer, limiting information processing only to one direction from the past to the future [11]. Another application utilizes the transformer model in RL as the replacement of convolutional layers for feature extraction. This is the case of the Swin Transformer model used for image processing [12]. It differs from the present paper, which does not incorporate the entire game screen as part of its input. The next application of the transformer model is in the temporal domain, but it only considers using the last timestep for action prediction, while the remaining timesteps are only used in the learning process to compute the model's error. It also uses a causal transformer [13].

Additional strategies for enhancing time-series prediction involve utilizing the last timestep, averaging features across timesteps, and determining the maximum value across timesteps.

The vision transformer [14] uses the features from the last timestep for action prediction, notably through its use of the class token. In the present paper, the last timestep represents the final state of the game, eliminating the necessity for an additional class token in the time series.

The average of the features across timesteps is used in the paper [15], and their maximum value is reported in [16]. The final alternative involves merging features over time, though this approach may result in a proliferation of inputs to the subsequent layer and contribute to overfitting in the model [17].

In contrast to previous research on Flappy Bird, the present paper aims to use the understanding of the temporal context from ray casting measurements. We employed the transformer model to process historical state measurements, subsequently aggregating these data in a judicious manner to forecast the current action of the agent. The sensor simulated in our study has a more restricted field of view when compared to the methods used in prior research [18]. Therefore, our model is designed to leverage its past knowledge of obstacles within the environment for effective navigation. Unlike neural network models that have been already applied to the Flappy Bird problem, our goal is to devise a method that can effectively condense information transmitted over time. This will allow us to express the qualities of actions for the current state of the agent and its corresponding response. Consequently, our model is designed to predict a categorical distribution of actions based on the current state of the agent, taking into account the agent's previously evaluated states. This approach allows the model to make informed decisions based on both the current and past states of the agent.

The key contributions of this paper are as follows:

- Improved performance: Our transformer model with a distance sensor significantly outperformed existing methods (an increase of over 50 times in both average and maximum scores). This suggests that real robots equipped with similar sensors can potentially achieve considerably higher accuracy when processing long sequences of sensor data.

- Sensor-focused learning: Unlike previous approaches, our agent solely relies on sensor data (not on the full game image) to learn from past experiences, identify obstacles, and navigate the environment. This suggests that focusing on relevant sensor data can be an efficient strategy for controlling robots.
- Visualizing and tracking the temporal similarity of sensor data: This research introduces a visualization technique to track similarities within sensor data sequences during a transformer's model training. This technique helps adjust the model to focus on the crucial measurements that impact the game's strategy and ultimate outcome, effectively discarding non-critical information. This approach was developed to reduce training times and lower memory requirements for the agent.
- Real-world applicability: Our findings have the potential to be applied to real robots operating in hazardous environments (comparable to the Flappy Bird simulation, where the agent can crash). By incorporating a "private zone" concept and deep learning guidance, robots could potentially navigate complex tasks while minimizing collisions and extending their operational lifespan.

This paper is organized as follows. Section 2 provides a review of the core algorithmic and computational approaches employed in this work. These include the dueling network architecture for Q-learning, the motion transformer architecture, the DeepMind Reverb database server used for machine learning, ray casting for obstacle detection, episodic memory incorporated into the transformer's input, and the private zone concept that aids in obstacle avoidance. Section 3 details the optimization process for the chosen methods and their hyperparameters. This section explores factors such as the number of timesteps used, the feature reduction techniques applied, and the optimal size of the private zone. It concludes with a crash analysis to assess the potential for enhancing the ultimate outcomes. Section 4 discusses future applications of this method and explores promising ways to improve it. Finally, Section 5 summarizes the key findings and conclusions presented throughout the paper.

2. Materials and Methods

The transformer model is trained by the dueling deep Q network approach. To achieve effective learning, we need to collect data on various paths explored within the state space and share the updated characteristics of our computational model. This task is facilitated by a specialized DeepMind Reverb database server. The state space only contains measurements from ray casting. The measurements from ray casting therefore warrant a dedicated exposition. Since the transformer is built on episodic memory, its usage in the Flappy Bird problem is independently addressed. Finally, an innovative approach involves the establishment of a private zone surrounding the agent to enhance its ability to maintain a secure distance while navigating obstacles. The introduction of this concept markedly improves performance throughout the learning process. A thorough analysis of these methodologies will be conducted in subsequent sections.

2.1. Dueling Deep Q Network

The principle of dueling network architecture is to extract features from the state space that are relevant for value function and advantage function prediction. The value function expresses how advantageous the current state of an agent is for its policy. The agent prioritizes traversing states that possess higher values. This strategy ensures the maximization of the overall value function. In order to make an informed selection among a multitude of potential actions, it is essential to ascertain the benefit associated with each action. This is achieved through the utilization of an advantage function [19]. In the case of a discrete action space, the probabilities of each action need to be expressed in the form of logits, which are predicted by a deep neural network model [20]. Logits represent Q-values, which can be computed according to the following relation [21]:

$$\boldsymbol{Q}(\boldsymbol{s},\boldsymbol{a}) = \boldsymbol{V}(\boldsymbol{s}) + (\boldsymbol{A}(\boldsymbol{s},\boldsymbol{a}) - \frac{1}{|\boldsymbol{A}|}\sum_{\boldsymbol{a}'}\boldsymbol{A}(\boldsymbol{s},\boldsymbol{a}')) \quad (1)$$

$\boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a})$ expresses the quality function for a given action $\boldsymbol{a}$ and in a given state $\boldsymbol{s}$. $\boldsymbol{V}(\boldsymbol{s})$ expresses the value function for a given state $\boldsymbol{s}$. $\boldsymbol{A}(\boldsymbol{s}, \boldsymbol{a})$ expresses the advantage function for a given action $\boldsymbol{a}$ in a given state $\boldsymbol{s}$. The average of the advantage function across actions in a given state $\underline{\boldsymbol{s}}$ is subtracted from the $\boldsymbol{A}(\boldsymbol{s}, \boldsymbol{a})$ function. Therefore, the advantage action has a zero mean [22].

The model is trained with the logarithmic hyperbolic cosine (LogCosh) error function, which is less sensitive to outliers than the more conventional mean squared error (MSE) function [23]. The error function of the model is expressed by the following:

$$\boldsymbol{L}(\boldsymbol{\theta}) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{U}(D)}\left[\boldsymbol{LogCosh}\left(\boldsymbol{y}^{\boldsymbol{DQN}} - \boldsymbol{Q}(\boldsymbol{s},\boldsymbol{a};\boldsymbol{\theta})\right)\right] \quad (2)$$

$$\boldsymbol{y}^{\boldsymbol{DQN}} = \boldsymbol{r} + \gamma\max_{\boldsymbol{a}'}\boldsymbol{Q}\left(\boldsymbol{s}',\boldsymbol{a}';\boldsymbol{\theta}^{-}\right) \quad (3)$$

$\boldsymbol{\mathcal{U}}(\boldsymbol{D})$ represents the uniform sampling from the replay buffer $\boldsymbol{D}$ that contains trajectories. $\boldsymbol{Q}(\boldsymbol{s},\boldsymbol{a};\boldsymbol{\theta})$ expresses the Q-value predicted by the model. The reward is symbolized by $\boldsymbol{r}$. $\boldsymbol{a}'$ is the next action expressed by the maximum Q-value in the next state $\boldsymbol{s}'$. $\boldsymbol{\theta}^{-}$ are parameters of the exponential moving average (EMA) model [24].

2.2. Motion Transformer

The motion transformer architecture is based on the encoder block in the transformer model [25]. The purpose of the encoder block is to traverse the input vector across the timeline in both directions. In this way, it is possible to look for relationships in historical data from past to future or from future to past and possibly associate them appropriately with the last timestep. The last timestep represents the source of information in the classical Markov decision process (MDP) [26]. A state vector representing the local memory of the model is fed to the model's input. The task of the model learning process is then to optimize the global memory (parameters) of the model so that the state space is ideally transformed into an action space. However, the output of the encoder block again represents a sequence; i.e., for each timestep, it predicts a set of extracted features from the input vector. Here, several methods are presented for extracting one particular distribution of the current action $\boldsymbol{a_t}$. One possible approach is to only use the extracted features from the last timestep to predict the distribution of actions $\boldsymbol{a_t}$, similarly to the class token [27]. The idea is to use the last timestep $\boldsymbol{s_t}$ to predict action $\boldsymbol{a_t}$ as in classical MDP. If some historical features are needed, they are inserted during the last timestep thanks to the attention mechanism. Another possibility is to use the average or maximum across all timesteps for each extracted feature separately.

Figure 1 shows the architecture of the motion transformer. The architecture consists of a preprocessing layer that adds position information to the input vector within the time series. This is followed by several encoder blocks that extract features along the time axis. The layer labeled X represents the reduction layer of the extracted features across the time series. Its type was varied during experiments. The last layers are value and advantage, representing fully connected output layers. Equation (1) is then applied to the output of the motion transformer.

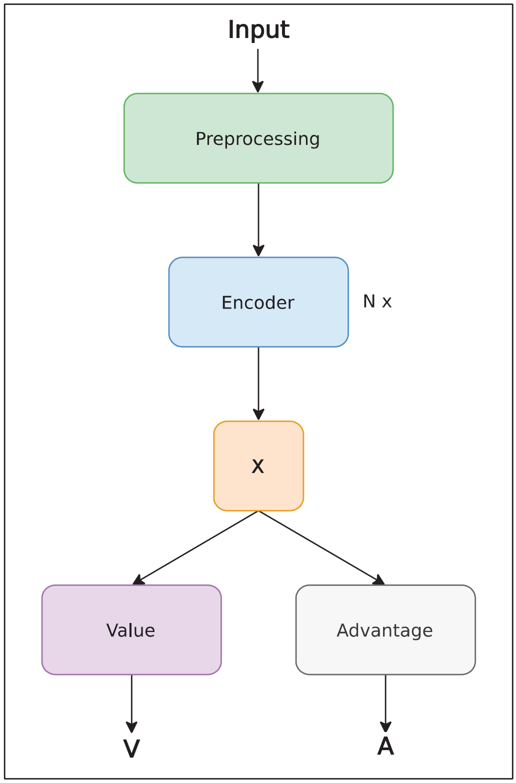


Figure 1. The architecture of the motion transformer model.

Figure 2 depicts the architecture of the preprocessing layer, which includes a projection layer in the form of a fully connected layer with a linear activation function. Its role is to transform the number of input features into the number of hidden features used in the rest of the model. Subsequently, a positional embedding, which is represented by trainable variables, is summed with the output of a fully connected layer. Thus, in this paper, embeddings for the time series are trained, along with the model [28].

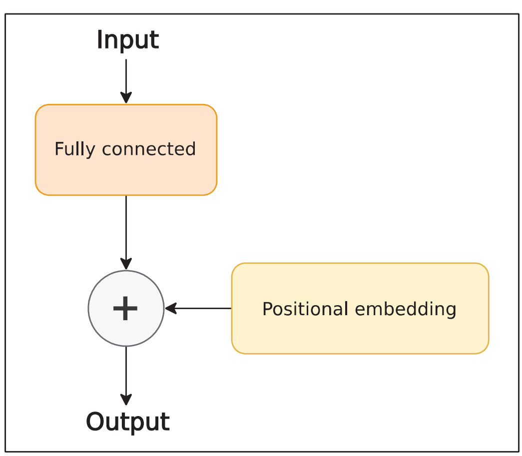


Figure 2. The architecture of the preprocessing layer.

The encoder block architecture is depicted in Figure 3. It consists of a pair of residual [29] sub-blocks. The first is multi-head attention [30], which processes the time series according to the following relations:

$$MultiHead(Q,K,V) = Concat(head_1, \ldots, head_n)W^O + b^O \quad (4)$$

$$head_i = Attention\left(QW_i^Q + b_i^Q, KW_i^K + b_i^K, VW_i^V + b_i^V\right) \quad (5)$$

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

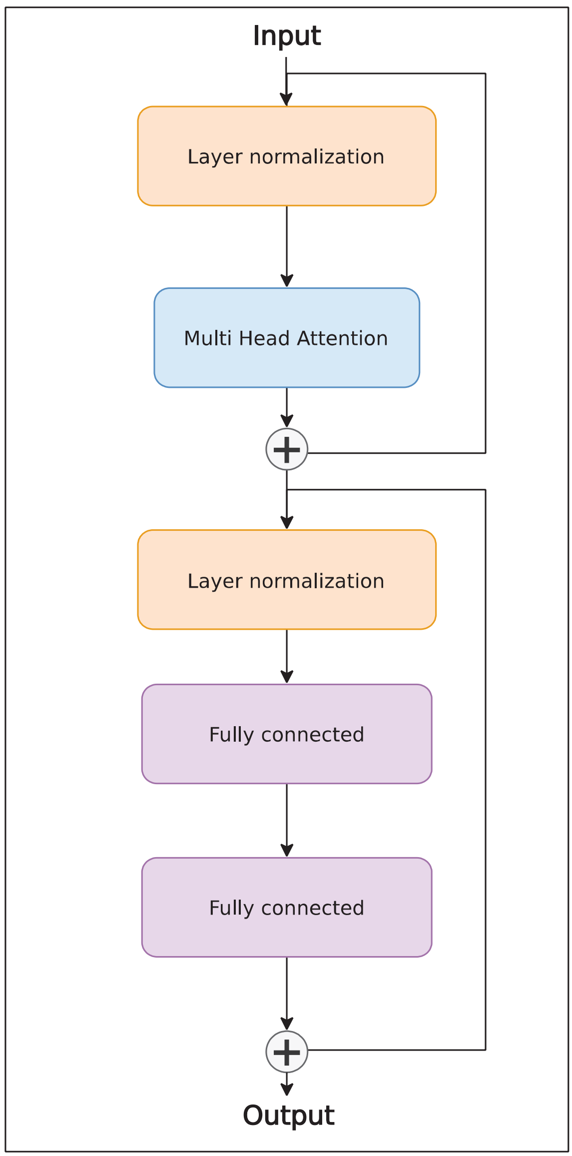


Figure 3. Architecture of encoder layer.

W^O represents weights, and b^O represents the biases of the linear transformation after merging heads. The process of merging heads comprises concatenating tensors along the head dimension (axis). W_i^Q, W_i^K, and W_i^V represent weights, and b_i^Q,b_i^K, and b_i^V represent biases of the linear transformation of the input vector of layer Q (Query), K (Key), and V (Value) into the space handled by the attention function. d_k represents the number of dimensions K after the linear projection of the layer input vector.

The second block is a multi-layer perceptron (MLP) [31], and its task is to nonlinearly transform the processed time series. The nonlinear activation function used is Gaussian error linear units (GeLUs) [32], applied after the first fully connected layer. It can be expressed by the following relation:

$$y = GeLU(xW_1 + b_1)W_2 + b_2 \tag{7}$$

W_1 and b_1 represent weight and bias parameters for the first fully connected layer to which the nonlinear transformation is subsequently applied. The parameters W_2 and b_2 represent the second layer of the block. This layer is responsible for executing a linear transformation on the output derived from the preceding layer. The dimension of this transformed output equals the dimension of the original input vector. Typically, the first layer of the block has 4 times more neurons than the last layer of the block [33].

2.3. Database Reverb

The DeepMind Reverb database server is used to effectively manage the collected trajectories and distribute the updated model parameters. This dedicated database server is tailored for RL algorithms where it acts as a replay buffer. Users can control strategies for selecting and removing elements from the database and options for controlling the ratio between sampled and inserted elements. The database server may contain several tables where trajectories or the parameters of the model are stored. An important feature is the compression of the stored data that the database server provides. In the case of overlapped trajectories, it is important to avoid storing duplicate trajectories [34]. The strategy used for sampling trajectories is the uniform sampler, which selects trajectories from the table with equal probability. The strategy for removing trajectories from a table is the first-in-first-out (FIFO) method. The ratio between sampled and inserted items is empirically set to 32 with 10% tolerance.

The client–server architecture used is illustrated in Figure 4. The server represents the database repository where trajectories are stored. The actor represents the client in the form of an agent, which gathers the experience in the form of trajectories through its interactions with the environment and stores the trajectories in the database server. The learner represents a client that retrieves trajectories from the database server and uses them to train an agent model. Following the training process, the agent receives the newly updated model parameters via the database server. A similar principle is used in the Acme framework [35].

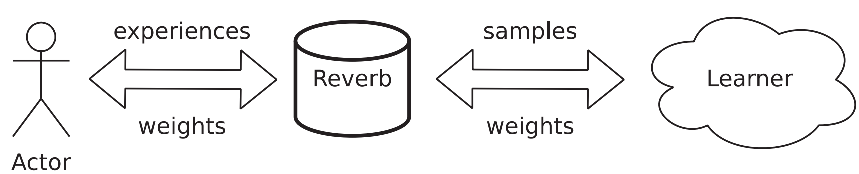


Figure 4. Client–server training.

2.4. LIDAR

The method used to detect nearby objects and track the agent's movement within the game environment employs a ray casting technique, specifically referred to as LIDAR for simplicity. It consists of 180 rays that are directed from the front of the agent to the right edge of the screen (see Figure 5). The endpoints of rays are determined by the following relations:

$$x = d_{MAX} * \cos\left(\alpha - player_\alpha - \frac{\pi}{2}\right) + player_x \tag{8}$$

$$y = d_{MAX} * \sin\left(\alpha - player_\alpha - \frac{\pi}{2}\right) + player_y \tag{9}$$

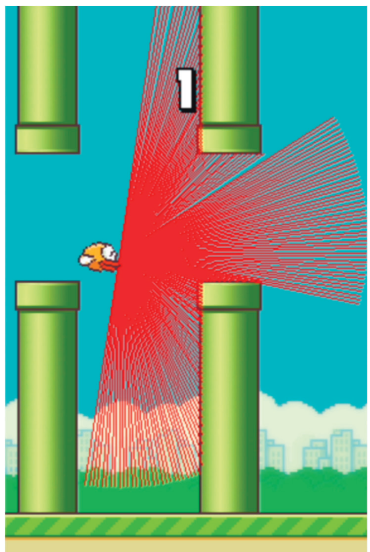


Figure 5. LIDAR sensor represented by ray casting.

d_{MAX} represents the maximum ray's length. The angle α determines the direction of ray radiation, and a $player_\alpha$ expresses the pitch angle of the player to the plane of the game space. The starting point of the ray is expressed by the coordinates of the front of the agents $player_x$ and $player_y$.

When the bird is pushed upward, it rotates towards the sky at a 45-degree angle. In the absence of player input, the bird slowly rotates towards the ground until reaching an angle of −90 degrees and then falls straight down.

The maximum ray length is the distance between the front of the agent and the right edge of the screen. Thus, the perpendicular ray touches the edge of the screen (if there are no obstacles) while other rays at a higher or lower angle usually do not reach the edge of the screen. This behavior mirrors the actual spread of ideal light and the measurements of its reflections from ideally reflective objects at different angles. In contrast, when other methods use an image generated by the game, the system sees it as a whole.

Here, rays emitted through ray casting spread out in a semicircular pattern and can detect obstacles within a limited area ahead of the agent. Moreover, if the bird is not positioned at the correct height and orientation relative to the game environment's plane, the detection rays do not even register the ground. Consequently, the agent cannot know its altitude throughout each episode. The sensor operates at an angular resolution of 1 degree and has a range limited only by the visible part of the environment ahead of the player. Collision with a ray occurs when the ray hits the surface of a pipe or the ground. The distance to the object is measured as the Euclidean distance between the agent's front, where the ray originates, and the collision point. However, these values are not statistically optimal for the model's input; hence, it is convenient to normalize them to the range [0, 1].

$$d_\alpha^{norm} = \frac{d_\alpha}{d_{MAX}} \tag{10}$$

d_α^{norm} represents the normalized distance to the object at an angle α. d_α expresses the measured value of the distance to the object. d_{MAX} represents the maximum ray's length. The d_{MAX} is defined as follows:

$$d_{MAX} = 0.8 * Screen_w - Player_w \tag{11}$$

$Player_w$ represents the width of the agent. The following measurements are given in pixels. The agent has a width of 34 and a height of 24. $Screen_w$ represents the width of the screen. The screen is the visible part of a game environment that can be seen when the game is rendered. The screen width is 288, and the height is 512.

2.5. Episodic Memory

The agent's state space consists of a fixed-length window of measurements from the timestep history. Therefore, it is necessary to create memory to store these measurements during a game episode. The entire contents of this memory serve as an input vector for the motion transformer. The first-in-first-out (FIFO) data structure ensures the flow of information in one direction, expressing the passage of time in the game environment. As new measurements are acquired during the episode, they replace the oldest ones in the queue. At the beginning of each episode, the queue is initialized with the initial state of the environment. While similarities are observed in the intended result relative to Atari stacking frames at the channel level [36], the present approach introduces a new timestep dimension in the input vector. This enables the model to exploit temporal relationships among measurements. The size of the memory determines how far back the model can effectively analyze measured states in the local history. Insufficient memory capacity can hinder information availability and impede effective action prediction. Conversely, excessively large memory unnecessarily drains computational resources.

Figure 6 illustrates the principle of applied episodic memory. The new state arrives at the end of the queue from the bottom. The oldest state leaves from the beginning of the queue, i.e., the top part. The input to the motion transformer represents all items that are stored in the queue and are ordered as they come in.

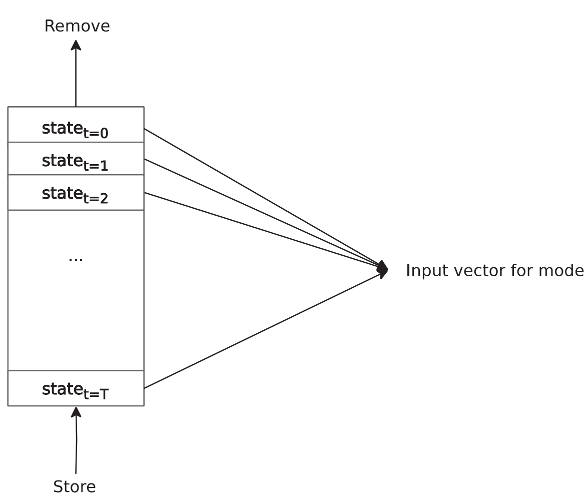


Figure 6. Architecture of episodic memory.

2.6. Private Zone around the Agent

In experiments, it was found that the agent tends to take risks and moves too close to the edges of the pipe when passing through the gap between pipes. There is a possibility of penalizing the agent for risky behavior in the policy. Therefore, a penalty for an obstacle approaching inside the agent's private zone was introduced. With this penalty in place, the agent is motivated to find the optimal solution for the given problem while considering its proximity to recognized obstacles. In a real-world application, object recognition would typically involve a dedicated deep neural network, which aims to distinguish between obstacles and desired objects, such as food or coins, in other gaming scenarios [37].

The agent needs to maintain a safe distance while navigating through obstacles to ensure its policy is not risky. This distance can be experimentally determined by finding the optimal radius for the private zone, which is represented by a circle. The circular model is chosen due to the sensor data being obtained in a circular polar grid format. As the rays are emitted from the agent's surface, the center of the private zone circle must coincide with the sensor's center. In the case where the simulation contains obstacles and objects the agent may need to interact with, such as collectibles, it is necessary to define this private zone dynamically. The classification process of obstacles vs. collectibles can be complex and involve sophisticated methods [38,39], and keeping the identified object between consecutive measurements can require specialized methods [40,41]. However, in the present game environment, where only obstacles exist, simple classification suffices.

$$\boldsymbol{r} = \frac{\boldsymbol{MAX}(\boldsymbol{Player_w}, \boldsymbol{Player_h}) + \boldsymbol{x}}{2} \tag{12}$$

The radius of the private zone circle is defined as $\boldsymbol{r}$, where $\boldsymbol{Player_w}$ represents the width of the agent and $\boldsymbol{Player_h}$ represents the height of the agent. Hyperparameter $\boldsymbol{x}$ specifies the size of the private zone. Since the rays radiate from ethe dges of the agent and not its center, when $\boldsymbol{x}$ is set to 0, the private zone's radius is equal to half of the agent's maximum dimension.

Figure 7 illustrates the agent's private zone, where the parameter $\boldsymbol{x}$ is set to 30. The gap between pipes is fixed at a size of 100 units (i.e., pixels), and the width of each pipe measures 52 units. As can be seen, a high value of $\boldsymbol{x}$ penalizes the agent if it attempts to navigate through the gap between pipes. Conversely, an $\boldsymbol{x}$ value that is too low diminishes the existence of a private zone, prompting the agent to take increased risks.

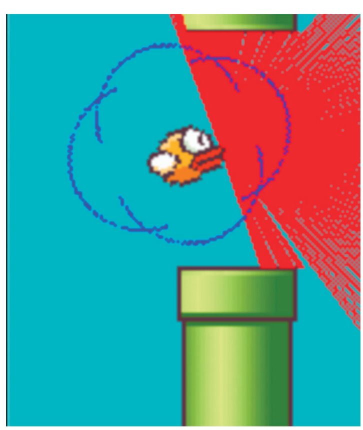


Figure 7. Agent's private zone.

3. Results

In order to improve the performance of the deep neural network controlling Flappy Bird's obstacle avoidance, various techniques required finetuning. This included selecting the right control system architecture and algorithmic techniques, as well as choosing appropriate hyperparameters during implementation. The following section provides detailed explanations of the key aspects of this process.

One aspect involved optimizing the number of timesteps retained in episodic memory. This determined the extent to which the agent would recall the short-term history and utilize it in its action predictions. Additionally, the study focused on refining the architecture of the model. Specifically, it explored whether it was effective to use the last timestep of the output series of actions. Alternatives included the global average or global maximum pooling. These operations involve computing the average or maximum of features across the timestep axis. These methods are commonly employed for reduction tasks, as seen in vision transformers and convolutional neural networks. Lastly, attention was directed towards determining the optimal size of the private zone, with options set to 30, 15, and 0.

The first tested configuration used 16 timesteps as the episodic memory size. Figure 8 shows the cosine similarity between embeddings for different pairs of timesteps. The closest similarities are in the region of the upper-left corner of the heatmap. Therefore, the subsequent experiment aimed to decrease the number of timesteps to diminish the density of similarities among timesteps and refine the optimal number of timesteps. Since there exists similarity among the initial timesteps, it is feasible to restrict their number. A total of 12 timesteps were used, which was anticipated to decrease the density of similarities, particularly in the upper-left corner.

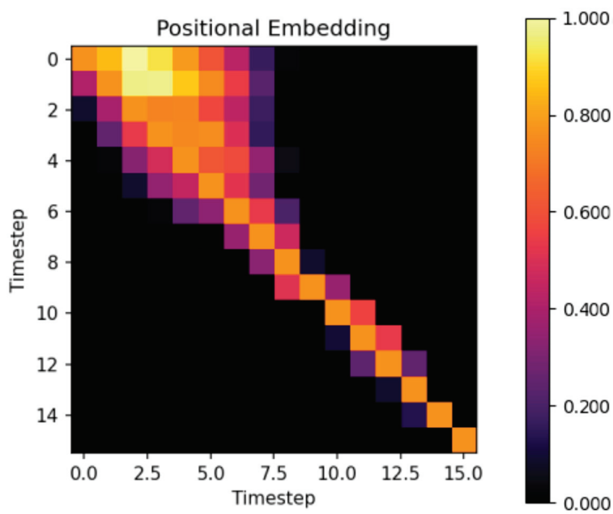


Figure 8. Similarity of timestep embeddings between 16 different timesteps.

The second experiment was to use only 12 timesteps. The cosine similarity between timestep embeddings is depicted in Figure 9. In contrast to using 16 timesteps, the density of timestep embedding similarities in the upper-left corner decreased, but the score of the agent did not significantly deteriorate. Figure 9 is not merely a subset of Figure 8; the difference in the density of similarities is apparent. Additionally, the similarity distribution is not perfectly symmetrical along the diagonal, with past steps showing more resemblance to future steps, especially for distant timeframes. This trend, however, is not observed in recent timesteps. Based on these observations, it could be beneficial in future experiments to explore using fewer past timesteps, as they exhibit similarities to future steps. Timesteps fewer than 12 or higher than 16 were not tested in this study.

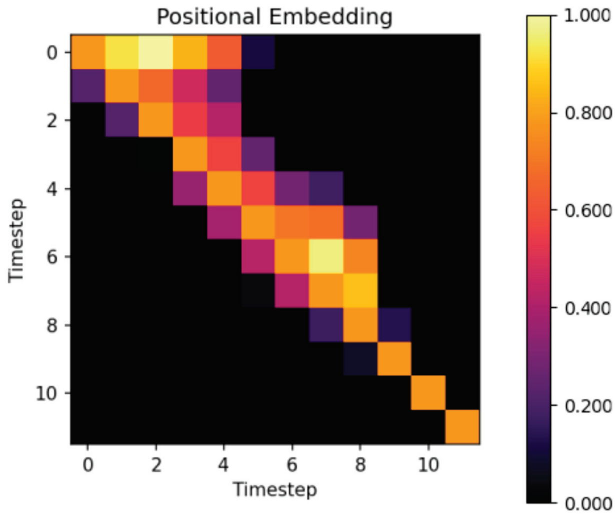


Figure 9. Similarity of timestep embeddings between 12 different timesteps.

In our investigation, we found that as the timestep increases, the similarity of embeddings decreases. This trend is particularly evident in the final timestep, regardless of whether there are 16 or 12 timesteps configured. Typically, the Markov decision process is applied only to the current state s_t. This implies that the most unique timestep must be the last timestep, which was also supported by measurement with both the 16 and 12 timestep configurations. Specifically, the last timestep exhibits the highest similarity only relative to itself. In this paper, a typical Markov decision process is modified. The historical states and current state are used simultaneously $s_{t\text{-}N:t}$, with the exception of the first state s_t due to its lack of existing history. Some historical states can probably have similar meanings for the agent. This adaptation draws parallels with word embedding, where words with similar meanings have a higher positive cosine similarity, but on the other hand, words with much different meanings have a small cosine similarity near zero [42].

In the following measurements, the comparison of the last timestep, global average, and global maximum pooling used data collected from 500 episodes. The average and maximum scores across episodes were measured for a deterministic, pre-trained agent. The score represents the number of pipes that the agent successfully passed through.

Table 1 shows the results of comparisons between different reduction techniques. The average of features along the timestep axis is significantly better than other approaches.

Table 1. Results of tested reduction methods.

Architecture	Timesteps	Highest Score	Average Score
Global average pooling	16	**2970**	324.198
Last timestep	16	2809	286.394
Global maximum pooling	16	1948	329.194
Global average pooling	12	2348	**380.284**
Last timestep	12	1922	335.114
Global maximum pooling	12	1128	152.858

The highest score is emphasized in boldface font.

Table 2 presents a comparison of the best results achieved in both the highest score and average score in this paper in contrast to other papers. This paper has significantly better scores.

Table 2. Caption.

Paper	Highest Score	Average Score
[43]	15	3.300
[4]	80	16.400
[6]	215	82.200
[5]	-	102.170
[7]	1491	209.298
This paper without a private zone	**2970**	**380.284**
This paper with a private zone	**74,755**	**13,156.590**

The score obtained in this paper is highlighted in boldface font.

Figures 10–12 show the tracking of the shifting pipelines along the timeline. The agent uses the history from 16 timesteps. A link to a video showing an animation of the changing attention matrix along with the changing environment is provided in the Supplementary Materials.

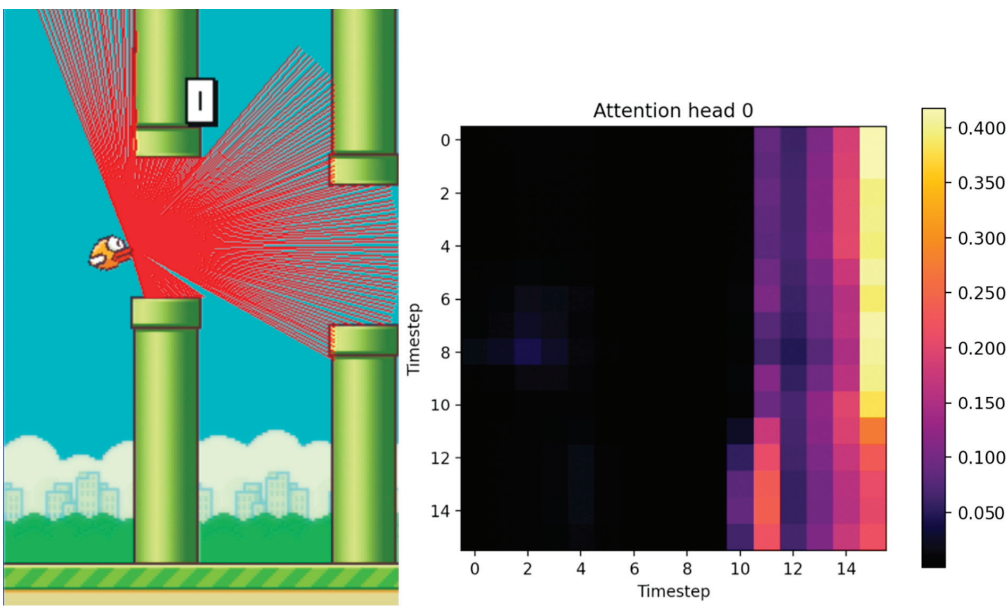


Figure 10. Agent with 16 timesteps entering the gap between the upper and lower pipes. (The brighter the color, the higher the attention value).

From the analysis of the agent's policy, it is evident that the agent takes risks and approaches the upper or lower pipes while passing through the gap between the pipes. To address this issue, it is necessary to designate a zone for the agent, beyond which, if obstacles are detected, the agent incurs a penalty of -0.5.

Likewise, the same -0.5 penalty stipulated in [5] for the agent reaching the top of the screen is also applied to the obstacles in the agent's private zone. In this game environment, all objects are regarded as obstacles.

Conversely, if the agent maintains a distance from the obstacles that is above a critical threshold, it is rewarded with a "still alive" reward valued at +0.1, similarly to [44].

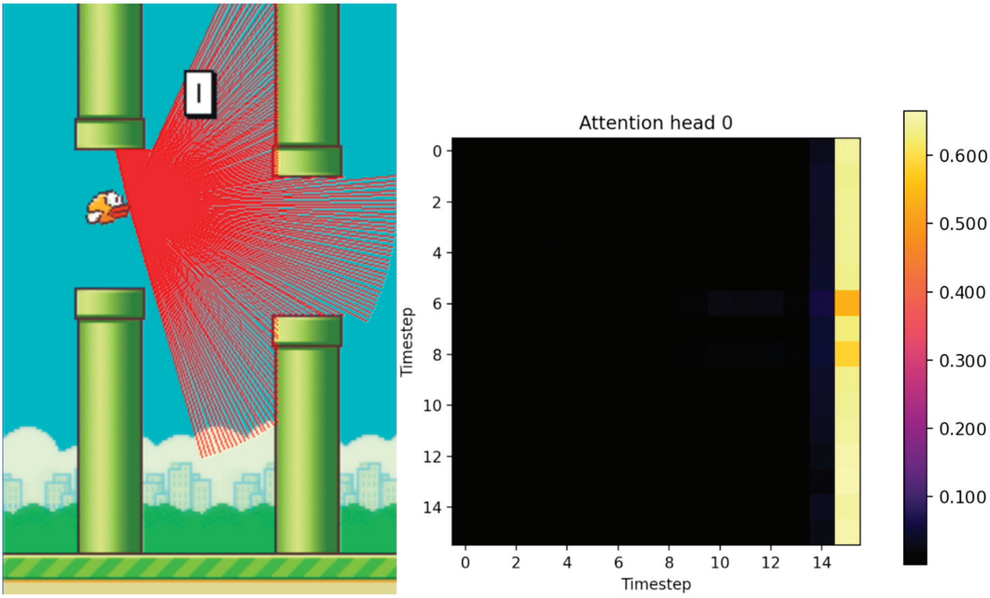


Figure 11. Agent with 16 timesteps in the gap between the upper and lower pipes. (The brighter the color, the higher the attention value).

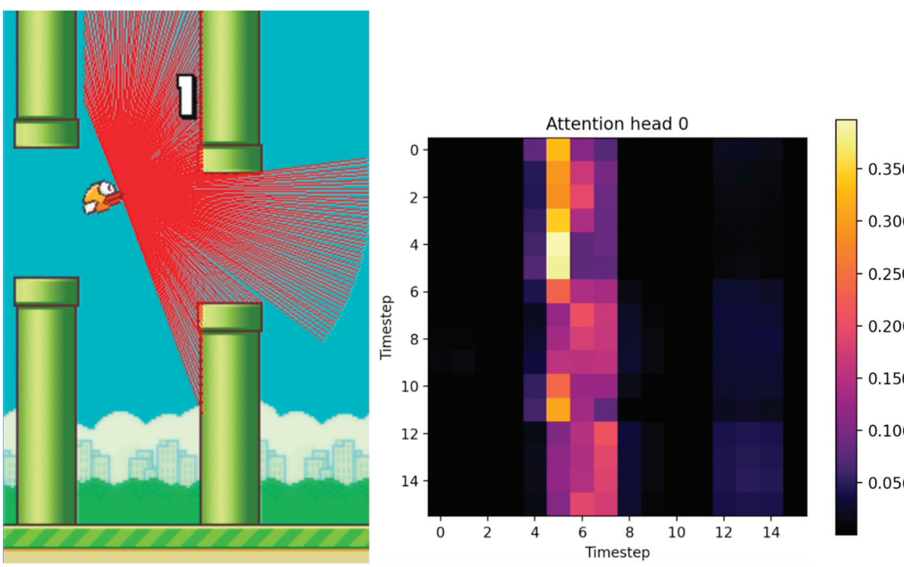


Figure 12. Agent with 16 timesteps passed the gap between the upper and lower pipes. (The brighter the color, the higher the attention value).

Figure 13 shows a histogram of the agent's score across various feature reduction techniques and private zone sizes. Experiments involving different feature reduction methods did not incorporate a penalty in the reward function for approaching obstacles too closely. Meanwhile, experiments with varying private zone sizes utilized global average pooling with 16 timesteps for feature reduction. Comparing the use of global maximum pooling to global average pooling, it is evident that the agent has a higher likelihood of scoring below 10 when employing the former method. The deep Q network generally overestimates the predicted Q-values [45]. Consequently, employing global maximum pooling may result in an overestimation of Q-values and a more risk-prone policy for the agent.

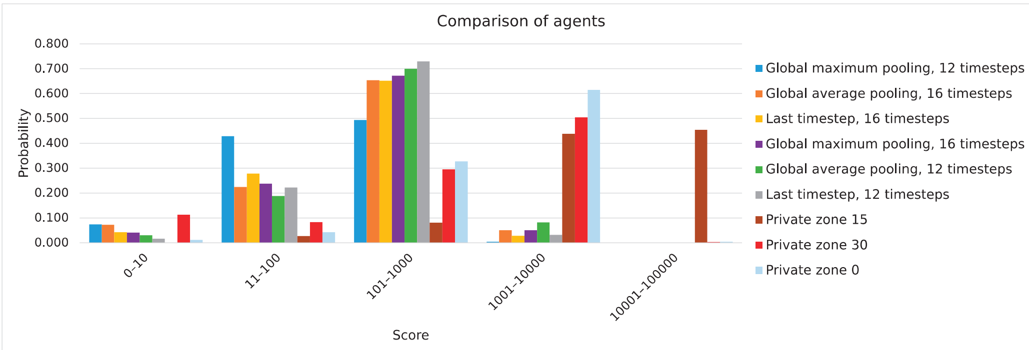


Figure 13. Histogram of score.

In the present study, it was observed that when only the last timestep was utilized, akin to the class token in the vision transformer [46], the global average pooling performed similarly to global maximum pooling.

The most stable control of Flappy Bird among the tested options of feature reduction was achieved via the global average pooling reduction method. It provided the highest maximum scores and average scores compared to the other methods of feature reduction. In contrast to global maximum pooling, global average pooling weighs down the activation by combining maximal and non-maximal activations [47]. This behavior leads to a reduction in the overestimation of Q-values predicted by the model and a less risky policy for the agent.

It was observed that using the optimal private zone size resulted in the agent achieving scores that were many times higher. The probability of obtaining a score of less than 100 was extremely low. Furthermore, a high probability of obtaining a score greater than 1000 was observed compared to agents without a private zone.

Table 3 presents a comparison between different private zone sizes. From a selection of several options, results indicate an optimal private zone size of 15. Excessively large values of private zone sizes would also penalize the agent for flying through the pipeline gap until it passes its center, which counts as a high positive reward of +1.0 to the exclusion of the other values of the reward function [48].

Table 3. Comparison of scores with different sizes of private zones.

Private Zone	Highest Score	Average Score
None	2970	380.284
0	10,250	2138.858
15	**74,755**	**13,156.590**
30	11,383	1645.654

Figure 14 presents the crash analysis for the collisions of the Flappy Bird without the private zone and with the optimal size private zone. While the score with the private zone is better by several orders of magnitude, the crush analysis shows that there still exists room for improvement. A robust solution should have an equal probability of hitting potential obstacles, while the results show that the Flappy Bird tends to crash almost exclusively into the bottom end of the upper pipe. The introduction of the private zone has minimized potential impact points, allowing future focus on suppressing these collisions. One approach to achieving this goal is to design a more robust reward function.

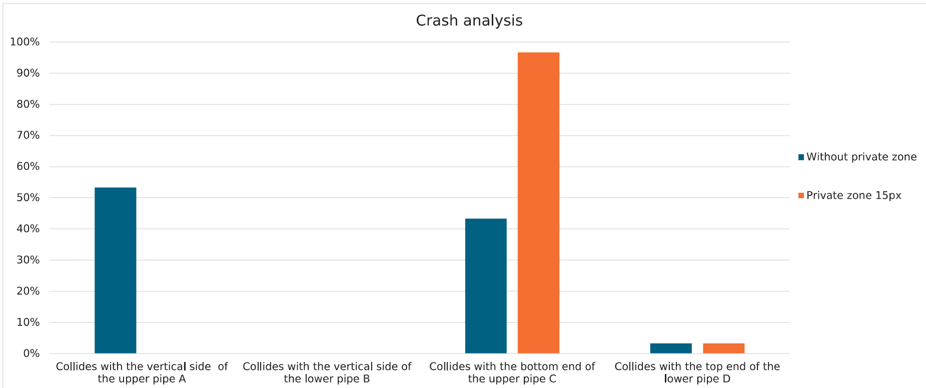


Figure 14. Crash analysis with and without the private zone.

Table 4 displays the hyperparameters used in all the experiments performed. Their values are set based on a combination of recommended settings. The recommended size of the replay buffer and the discount factor are taken from [49]. The multiplier of the MLP block dimension, type of learning rate schedule, and gradient clipping are based on [50]. A private zone with a value of None indicates the absence of a penalty rule for approaching obstacles in the reward function. The other numerical values of the private zone size express the x of Equation (12).

Table 4. Hyperparameters of the agent and learner model.

Hyperparameter	Description	Value
port	Database server port	8000
max_replay_size	Maximum database memory	1,000,000
samples_per_insert	Samples per insert ratio for reverb	32
temp_init	Initial Boltzmann temperature for exploration	0.500
temp_min	Minimal Boltzmann temperature	0.010
temp_decay	Decay of Boltzmann temperature	0.999999
warmup_steps	Warmup steps for learning rate cosine scheduler	1000
train_steps	Training steps	1,000,000
batch_size	Batch size	256
gamma	Discount factor	0.990
tau	Tau factor (for EMA model)	0.005
num_layers	Num. of encoder blocks	2
embed_dim	Embedding dimension	128
ff_mult	Multiplier of MLP block dimension	4
num_heads	Num. of attention heads	6
learning_rate	Learning rate	3×10^{-4}
global_clipnorm	Globally normalized clipping of gradient	1
weight_decay	Weight decay for AdamW optimizer	1×10^{-4}
frame_stack	Size of short-term (episodic) memory	16 or 12
player_private_zone	Size of agent's private zone	None, 0, 15 or 30

4. Discussion

Utilizing a transformer neural network to control a simulated agent via ray casting as a simple LIDAR sensor has potentially diverse applications across several domains. Remote sensing technology integrated with advanced AI-based control can be beneficial in the following contexts:

In virtual reality and games, avatars or characters can benefit from more natural and responsive interactions.

The method for improving navigation using ray casting in 2D could potentially be expanded to utilize true LIDAR in 3D space. In the future, this could lead to advancements in robotics; autonomous vehicles like self-driving cars, drones, or any kind of mobile robots that require effective navigation; and obstacle avoidance capabilities. In disaster-stricken areas, such robots can aid in search and rescue missions. Enhanced agents can streamline tasks such as inventory management and material handling in warehouses. Additionally, robotic arms could better manipulate objects in dynamic environments.

In each of these contexts, the integration of ray casting and transformer neural network control should enable the agent to make informed decisions based on temporal and spatial information.

When considering the selection of algorithmic procedures and hyperparameters, there exists ample room for exploration and experimentation with various possibilities.

When storing high-dimensional states in episodic memory, it would be more convenient to only extract the significant features for storage. For this purpose, an AutoEncoder-type model could be used to compress the input vector.

Another consideration is the initialization of episodic memory. Currently, it duplicates the initial state, but one alternative includes creating an embedding for the empty state containing episodic memory at each episode's start. The next option is to dynamically adjust the number of timesteps with respect to the input to the motion transformer, while ensuring the proper assignment of positional embedding for incrementing states from the timesteps.

A promising research direction is exploring the impact of cosine similarity on the optimal number of timesteps. This involves investigating whether the similarity of timestep embeddings can reduce the necessary number of timesteps. Further study is needed to validate the effect of positional embedding in reducing timesteps across various game environments.

In the case of the Flappy Bird game, future research should also try the possibility of adding a weighted reward for keeping a safe distance from the upper pipe more than from other obstacles. This research direction follows from the results of the error analysis. The subject of further research is also to study and rectify failures after the agent has performed a very large number of steps in the environment. Potential improvements might be anticipated in algorithms based on the deep Q Learning principle, such as dueling deep Q learning or double deep Q learning. Numerical instability should also be checked, as well as more advanced Actor–Critic-type models such as A2C or PPO.

Further improvements could be attained by establishing a dynamic private zone around the agent. The private zone could be delineated by deep neural network prediction, whether objects crossing the private zone boundary are obstacles or aids in achieving a specific task. Such a model could directly adjust the complex reward function necessary for task completion without exposing the agent to risky behavior.

5. Conclusions

Our study presents novel guidance control using LIDAR sensors represented by the ray casting method for obstacle detection and agent navigation within obstacle-filled environments. The designed motion transformer model effectively grasped the temporal dynamics between sensor readings. The findings demonstrate the model's ability to adaptively respond to the agent's movement among pipelines, as reflected in the attention matrix. The model's attention mechanism prioritizes past or present sensor data, or a combination thereof, based on the spatial distribution of pipelines in the surroundings. Additionally, the results show that employing average reduction techniques helps mitigate the risk of overestimating Q values. Furthermore, the incorporation of a private zone for the agent contributes to the formulation of a less risky navigation policy.

In this paper, the average score (the number of passes through pipeline gaps) obtained by the agent without a private zone is 182 percent better compared to the best results obtained by the competitors. The highest score achieved by the agent without a private zone compared to the best competitors' obtained results is 199 percent better. The agent with a private zone of 15 pixels achieved an average score that was 6286 percent better than the best competitors' average agent score and a maximum score that was 5014 percent better than the competition's best results in terms of maximum agent score.

Supplementary Materials: The following supporting information can be downloaded at the following: interactive charts: https://wandb.ai/markub/rl-toolkit/groups/FlappyBird-v0 (accessed on 16 October 2023); source codes: https://github.com/markub3327/rl-toolkit (accessed on 16 October 2023) and https://github.com/markub3327/flappy-bird-gymnasium (accessed on 16 October 2023); YouTube video: https://youtu.be/aZQxuDCyHoI (accessed on 22 December 2023).

Author Contributions: Conceptualization, I.D.L. and J.P.; methodology, M.K.; software, M.K.; validation, M.K.; formal analysis, J.P.; investigation, M.K.; resources I.D.L.; data curation, M.K.; writing—original draft preparation, M.K.; writing—review and editing, J.P.; visualization, M.K.; supervision, I.D.L. and J.P.; project administration, I.D.L.; funding acquisition, I.D.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Cultural and Educational Grant Agency MŠVVaŠ SR, grant number KEGA 020UCM-4/2022, and by Erasmus+ project FAAI: The Future is in Applied Artificial Intelligence—2022-1-PL01-KA220-HED-000088359, work package WP4.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (accessed on 10 December 2023).
2. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are transformers effective for time series forecasting? *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 11121–11128. Available online: https://ojs.aaai.org/index.php/AAAI/article/view/26317/26089 (accessed on 10 December 2023). [CrossRef]
3. Wei, S. Reinforcement Learning for Improving Flappy Bird Game. *Highlights Sci. Eng. Technol.* **2023**, *34*, 244–249. Available online: https://drpress.org/ojs/index.php/HSET/article/download/5479/5298 (accessed on 10 December 2023). [CrossRef]
4. Pilcer, L.S.; Hoorelbeke, A.; Andigne, A.D. Playing Flappy Bird with Deep Reinforcement Learning. *IEEE Trans. Neural Netw.* **2015**, *16*, 285–286. Available online: https://www.researchgate.net/profile/Louis-Samuel-Pilcer/publication/324066514_Playing_Flappy_Bird_with_Deep_Reinforcement_Learning/links/5abbc2230f7e9bfc045592df/Playing-Flappy-Bird-with-Deep-Reinforcement-Learning.pdf (accessed on 10 December 2023).
5. Yang, K. Using DQN and Double DQN to Play Flappy Bird. In Proceedings of the 2022 International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID 2022), Xi'an, China, 15–17 April 2022; Atlantis Press: Amsterdam, The Netherlands, 2022; pp. 1166–1174. Available online: https://www.atlantis-press.com/article/125977189.pdf (accessed on 10 December 2023).
6. Chen, K. Deep Reinforcement Learning for Flappy Bird. CS 229 Machine-Learning Final Projects. 2015. Available online: https://cs229.stanford.edu/proj2015/362_report.pdf (accessed on 10 December 2023).
7. Vu, T.; Tran, L. FlapAI Bird: Training an Agent to Play Flappy Bird Using Reinforcement Learning Techniques. *arXiv* **2020**, arXiv:2003.09579.
8. Li, J.; Yin, Y.; Chu, H.; Zhou, Y.; Wang, T.; Fidler, S.; Li, H. Learning to generate diverse dance motions with transformer. *arXiv* **2020**, arXiv:2008.08171.
9. Shi, S.; Jiang, L.; Dai, D.; Schiele, B. Motion transformer with global intention localization and local movement refinement. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 6531–6543. Available online: https://proceedings.neurips.cc/paper_files/paper/2022/file/2ab47c960bfee4f86dfc362f26ad066a-Paper-Conference.pdf (accessed on 10 December 2023).
10. Hu, M.; Zhu, X.; Wang, H.; Cao, S.; Liu, C.; Song, Q. STDFormer: Spatial-Temporal Motion Transformer for Multiple Object Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 6571–6594. Available online: https://ieeexplore.ieee.org/iel7/76/4358651/10091152.pdf (accessed on 10 December 2023). [CrossRef]
11. Esslinger, K.; Platt, R.; Amato, C. Deep Transformer Q-Networks for Partially Observable Reinforcement Learning. *arXiv* **2022**, arXiv:2206.01078.
12. Meng, L.; Goodwin, M.; Yazidi, A.; Engelstad, P. Deep Reinforcement Learning with Swin Transformer. *arXiv* **2022**, arXiv:2206.15269.

13. Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; Mordatch, I. Decision transformer: Reinforcement learning via sequence modeling. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15084–15097. Available online: https://proceedings.neurips.cc/paper_files/paper/2021/file/7f489f642a0ddb10272b5c31057f0663-Paper.pdf (accessed on 10 December 2023).
14. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
15. Liu, R.; Ji, C.; Niu, J.; Guo, B. Research on intrusion detection method based on 1D-ICNN-BiGRU. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2022; Volume 2347, p. 012001. Available online: https://iopscience.iop.org/article/10.1088/1742-6596/2347/1/012001/pdf (accessed on 10 December 2023).
16. Crocioni, G.; Pau, D.; Delorme, J.M.; Gruosso, G. Li-ion batteries parameter estimation with tiny neural networks embedded on intelligent IoT microcontrollers. *IEEE Access* **2020**, *8*, 122135–122146. Available online: https://ieeexplore.ieee.org/iel7/6287639/6514899/09133084.pdf (accessed on 10 December 2023). [CrossRef]
17. Gholamalinezhad, H.; Khosravi, H. Pooling Methods in Deep Neural Networks, a Review. *arXiv* **2020**, arXiv:2009.07485.
18. Anders, K.; Winiwarter, L.; Lindenbergh, R.; Williams, J.G.; Vos, S.E.; Höfle, B. 4D objects-by-change: Spatiotemporal segmentation of geomorphic surface change from LiDAR time series. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 352–363. Available online: https://www.sciencedirect.com/science/article/pii/S0924271619302850 (accessed on 10 December 2023). [CrossRef]
19. Wang, Z.; Schaul, T.; Hessel, M.; Hasselt, H.; Lanctot, M.; Freitas, N. Dueling network architectures for deep reinforcement learning. In Proceedings of the International Conference on Machine Learning, PMLR, New York City, NY, USA, 19–24 June 2016; pp. 1995–2003. Available online: http://proceedings.mlr.press/v48/wangf16.pdf (accessed on 10 December 2023).
20. Haarnoja, T.; Tang, H.; Abbeel, P.; Levine, S. Reinforcement learning with deep energy-based policies. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 1352–1361. Available online: http://proceedings.mlr.press/v70/haarnoja17a/haarnoja17a.pdf (accessed on 10 December 2023).
21. Peng, B.; Sun, Q.; Li, S.E.; Kum, D.; Yin, Y.; Wei, J.; Gu, T. End-to-end autonomous driving through dueling double deep Q-network. *Automot. Innov.* **2021**, *4*, 328–337. Available online: https://link.springer.com/content/pdf/10.1007/s42154-021-00151-3.pdf (accessed on 10 December 2023). [CrossRef]
22. Liu, F.; Li, S.; Zhang, L.; Zhou, C.; Ye, R.; Wang, Y.; Lu, J. 3DCNN-DQN-RNN: A deep reinforcement learning framework for semantic parsing of large-scale 3D point clouds. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 5678–5687. Available online: https://openaccess.thecvf.com/content_ICCV_2017/papers/Liu_3DCNN-DQN-RNN_A_Deep_ICCV_2017_paper.pdf (accessed on 10 December 2023).
23. Saleh, R.A.; Saleh, A.K. Statistical Properties of the Log-Cosh Loss Function Used in Machine Learning. *arXiv* **2022**, arXiv:2208.04564.
24. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1195–1204. Available online: https://proceedings.neurips.cc/paper/2017/file/68053af2923e00204c3ca7c6a3150cf7-Paper.pdf (accessed on 10 December 2023).
25. Tummala, S.; Kadry, S.; Bukhari, S.A.C.; Rauf, H.T. Classification of brain tumor from magnetic resonance imaging using vision transformers ensembling. *Curr. Oncol.* **2022**, *29*, 7498–7511. Available online: https://www.mdpi.com/1718-7729/29/10/590/htm (accessed on 10 December 2023). [CrossRef]
26. Wang, X.; Yang, Z.; Chen, G.; Liu, Y. A Reinforcement Learning Method of Solving Markov Decision Processes: An Adaptive Exploration Model Based on Temporal Difference Error. *Electronics* **2023**, *12*, 4176. Available online: https://www.mdpi.com/2079-9292/12/19/4176 (accessed on 10 December 2023). [CrossRef]
27. Feng, H.; Yang, B.; Wang, J.; Liu, M.; Yin, L.; Zheng, W.; Yin, Z.; Liu, C. Identifying malignant breast ultrasound images using ViT-patch. *Appl. Sci.* **2023**, *13*, 3489. Available online: https://www.mdpi.com/2076-3417/13/6/3489 (accessed on 10 December 2023). [CrossRef]
28. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 770–778. Available online: https://openaccess.thecvf.com/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf (accessed on 10 December 2023).
30. Hasan, F.; Huang, H. MALS-Net: A multi-head attention-based LSTM sequence-to-sequence network for socio-temporal interaction modelling and trajectory prediction. *Sensors* **2023**, *23*, 530. Available online: https://www.mdpi.com/1424-8220/23/1/530/pdf (accessed on 10 December 2023). [CrossRef]
31. Mogan, J.N.; Lee, C.P.; Lim, K.M.; Muthu, K.S. Gait-ViT: Gait Recognition with Vision Transformer. *Sensors* **2022**, *22*, 7362. Available online: https://www.mdpi.com/1424-8220/22/19/7362/pdf (accessed on 10 December 2023). [CrossRef]
32. Hendrycks, D.; Gimpel, K. Gaussian Error Linear Units (Gelus). *arXiv* **2016**, arXiv:1606.08415.
33. Sun, W.; Wang, H.; Xu, J.; Yang, Y.; Yan, R. Effective Convolutional Transformer for Highly Accurate Planetary Gearbox Fault Diagnosis. *IEEE Open J. Instrum. Meas.* **2022**, *1*, 1–9. Available online: https://ieeexplore.ieee.org/iel7/9552935/9775186/09828477.pdf (accessed on 10 December 2023). [CrossRef]

34. Cassirer, A.; Barth-Maron, G.; Brevdo, E.; Ramos, S.; Boyd, T.; Sottiaux, T.; Kroiss, M. Reverb: A Framework for Experience Replay. *arXiv* **2021**, arXiv:2102.04736.
35. Hoffman, M.W.; Shahriari, B.; Aslanides, J.; Barth-Maron, G.; Momchev, N.; Sinopalnikov, D.; Stańczyk, P.; Ramos, S.; Raichuk, A.; Vincent, D.; et al. Acme: A Research Framework for Distributed Reinforcement Learning. *arXiv* **2020**, arXiv:2006.00979.
36. Lapan, M. *Deep Reinforcement Learning Hands-On: Apply Modern RL Methods, with Deep Q-Networks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More*; Packt Publishing Ltd.: Birmingham, UK, 2018.
37. Singh, A.; Yang, L.; Hartikainen, K.; Finn, C.; Levine, S. End-to-End Robotic Reinforcement Learning without Reward Engineering. *arXiv* **2019**, arXiv:1904.07854.
38. Capellier, E.; Davoine, F.; Cherfaoui, V.; Li, Y. Evidential deep learning for arbitrary LIDAR object classification in the context of autonomous driving. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1304–1311. Available online: https://hal.science/hal-02322434/file/IV19-Edouard.pdf (accessed on 10 December 2023).
39. Skrinárová, J.; Huraj, L.; Siládi, V. A neural tree model for classification of computing grid resources using PSO tasks scheduling. *Neural Netw. World* **2013**, *23*, 223. Available online: https://www.proquest.com/docview/1418215646/fulltextPDF/AF8F42E64A49412CPQ/1?accountid=49441&sourcetype=Scholarly%20Journals (accessed on 10 December 2023). [CrossRef]
40. Sualeh, M.; Kim, G.W. Dynamic multi-lidar based multiple object detection and tracking. *Sensors* **2019**, *19*, 1474. Available online: https://www.mdpi.com/1424-8220/19/6/1474/pdf (accessed on 10 December 2023). [CrossRef]
41. Kyselica, D.; Šilha, J.; Ďurikovič, R.; Bartková, D.; Tóth, J. Towards image processing of reentry event. *J. Appl. Math. Stat. Inform.* **2023**, *19*, 47–60. Available online: https://sciendo.com/article/10.2478/jamsi-2023-0003 (accessed on 10 December 2023). [CrossRef]
42. Orkphol, K.; Yang, W. Word sense disambiguation using cosine similarity collaborates with Word2vec and WordNet. *Future Internet* **2019**, *11*, 114. Available online: https://www.mdpi.com/1999-5903/11/5/114/pdf (accessed on 10 December 2023). [CrossRef]
43. Appiah, N.; Vare, S. Playing Flappy Bird with Deep Reinforcement Learning. 2018. Available online: http://vision.stanford.edu/teaching/cs231n/reports/2016/pdfs/111_Report.pdf (accessed on 10 December 2023).
44. Li, L.; Jiang, Z.; Yang, Z. Playing Modified Flappy Bird with Deep Reinforcement Learning. 2023. Available online: https://github.com/SeVEnMY/DeepLearningFinal (accessed on 10 December 2023).
45. Hasselt, H. Double Q-Learning. *Adv. Neural Inf. Process. Syst.* **2010**, *23*, 2613–2621. Available online: https://proceedings.neurips.cc/paper_files/paper/2010/file/091d584fced301b442654dd8c23b3fc9-Paper.pdf (accessed on 10 December 2023).
46. Al Rahhal, M.M.; Bazi, Y.; Jomaa, R.M.; AlShibli, A.; Alajlan, N.; Mekhalfi, M.L.; Melgani, F. COVID-19 detection in Ct/X-ray imagery using vision transformers. *J. Pers. Med.* **2022**, *12*, 310. Available online: https://www.mdpi.com/2075-4426/12/2/310 (accessed on 10 December 2023). [CrossRef] [PubMed]
47. Passricha, V.; Aggarwal, R.K. A comparative analysis of pooling strategies for convolutional neural network based Hindi ASR. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 675–691. Available online: https://link.springer.com/article/10.1007/s12652-019-01325-y (accessed on 10 December 2023). [CrossRef]
48. Mazumder, S.; Liu, B.; Wang, S.; Zhu, Y.; Yin, X.; Liu, L.; Li, J.; Huang, Y. Guided Exploration in Deep Reinforcement Learning. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019; Available online: https://openreview.net/forum?id=SJMeTo09YQ (accessed on 10 December 2023).
49. Hessel, M.; Modayil, J.; Van Hasselt, H.; Schaul, T.; Ostrovski, G.; Dabney, W.; Horgan, D.; Piot, B.; Azar, M.; Silver, D. Rainbow: Combining improvements in deep reinforcement learning. *AAAI Conf. Artif. Intell.* **2018**, *32*, 1. Available online: https://ojs.aaai.org/index.php/AAAI/article/download/11796/11655 (accessed on 10 December 2023). [CrossRef]
50. Bao, H.; Dong, L.; Piao, S.; Wei, F. Beit: Bert Pre-Training of Image Transformers. *arXiv* **2021**, arXiv:2106.08254.

sensors

Article

Enhancing Smart Building Surveillance Systems in Thin Walls: An Efficient Barrier Design

Taewoo Lee and Hyunbum Kim *

Department of Embedded Systems Engineering, Incheon National University, Incheon 22012, Republic of Korea; hst0092@inu.ac.kr
* Correspondence: hyunbumkim@ieee.org; Tel.: +82-32-835-8764

Abstract: This paper introduces an efficient barrier model for enhancing smart building surveillance in harsh environment with thin walls and structures. After the main research problem of minimizing the total number of wall-recognition surveillance barriers, we propose two distinct algorithms, Centralized Node Deployment and Adaptation Node Deployment, which are designed to address the challenge by strategic placement of surveillance nodes within the smart building. The Centralized Node Deployment aligns nodes along the thin walls, ensuring consistent communication coverage and effectively countering potential disruptions. Conversely, the Adaptation Node Deployment begins with random node placement, which adapts over time to ensure efficient communication across the building. The novelty of this work is in designing a novel barrier system to achieve energy efficiency and reinforced surveillance in a thin-wall environment. Instead of a real environment, we use an ad hoc server for simulations with various scenarios and parameters. Then, two different algorithms are executed through those simulation environments and settings. Also, with detailed discussions, we provide the performance analysis, which shows that both algorithms deliver similar performance metrics over extended periods, indicating their suitability for long-term operation in smart infrastructure.

Keywords: smart building; surveillance; infrastructure; walls; efficiency

Citation: Lee, T.; Kim, H. Enhancing Smart Building Surveillance Systems in Thin Walls: An Efficient Barrier Design. *Sensors* **2024**, *24*, 595. https://doi.org/10.3390/s24020595

Academic Editors: Hossam A. Gabbar, Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 13 December 2023
Revised: 12 January 2024
Accepted: 15 January 2024
Published: 17 January 2024

1. Introduction

In the era of smart cities, the concept of smart buildings has emerged as a vital component of urban infrastructure to provide convenient lives to citizens. In the realm of smart buildings, seamless communication and continuous surveillance between various components of the infrastructure are crucial. One of the primary obstacles in establishing such communication is the presence of physical barriers like walls—structures which can significantly disrupt data flow, diminish detection accuracy, and affect the system's overall performance. These buildings are characterized by their integration of IoT (Internet of Things) technologies, facilitating advanced automation and real-time control over various subsystems, such as energy management, security, and environmental controls [1–8]. Also, it is highly anticipated that 5G and 6G communication technologies are utilized in smart buildings and expansive spaces [9–15].

However, the increase in smart buildings and their effective operations depend on the availability and efficiency of communication systems in these structures. One of the most significant challenges faced in deploying effective communication systems within smart buildings is the presence of physical barriers, primarily walls, which can significantly disrupt the flow of data. These result in coverage gaps and reductions in the overall performance of the communication system. This issue becomes exacerbated when we deliberate on solid walls, which present a formidable obstacle to the propagation of electromagnetic waves, leading to disconnected spaces within the building and jeopardizing the optimal functioning of the smart system.

On the other hand, surveillance and secure monitoring are considered as critical tasks in smart buildings; these tasks can be supported by heterogeneous mobile robots, drones, UAVs (Unmanned Aerial Vehicles), IoT devices, and intelligent components [16–22]. To provide reinforced surveillance and secure monitoring, the concept of a barrier can be applied to smart buildings because the formation of a barrier and its creation guarantee that any penetrations or mobile objects are detected by system members in the built surveillance barriers within the requested three-dimensional space and plane area [23–29]. Essentially, it has been known that the barrier has been used for numerous applications, including virtual emotion surveillance, digital twins, maritime transportation systems, public and private areas, patrol services, border surveillance, smart complex surveillance, virtual emotion informatics, and geographic segmentation surveillance [30–34]. However, it is not sufficient to form secure building surveillance in smart buildings with the walls in order to deliberate on energy-efficient formations when the walls are present because these may affect the detection and the communication by the deployed system members or components equipped with wireless transmitters and receivers. Thus, it is highly necessary to proceed by handling the issue so as to enhance secure surveillance in smart buildings with consideration of energy efficiency.

To solve this issue, this paper proposes an approach: the preemptive placement of communication nodes within thin walls. The idea is to leverage the thin walls, seen as obstacles, as conduits for communication. By strategically embedding nodes within these walls and positioning additional nodes in the adjacent spaces, a communication barrier can be created. This barrier bridges the disconnect caused by the physical walls, thereby enabling seamless data transfer between spaces.

The objective function of this study is to pursue energy-efficient surveillance in smart buildings within the walls. It follows that the main task in the proposed system is to minimize the number of nodes or system members while ensuring optimal communication with surveillance. This consideration is important, as an excessive number of nodes can lead to higher costs, increased energy consumption, and potential signal interference. The potential for signal interference also rises with an increase in the number of nodes. As more nodes are placed in proximity, there is a higher likelihood of overlapping signals, which can lead to data corruption and a drop in network performance [8]. Therefore, the balance between the number of nodes and communication effectiveness is a complex challenge that requires a careful and innovative approach to ensure that smart buildings can function efficiently without incurring prohibitive costs.

Based on the above motivations, the main contributions of the paper are summarized below:

- First, we design an efficient barrier system for enhancing smart building surveillance in harsh environments with walls and infrastructure. The proposed system is designed to consider energy-efficient surveillance and optimal formations of system components;
- Then, this paper presents a formal definition of the research problem to minimize the number of nodes or system components, in order to ensure secure surveillance and communication among system components;
- To resolve the problem, we propose two different algorithms for the preemptive placement of nodes within thin walls and the adjacent spaces. These algorithms aim to minimize the number of nodes to an optimal level and to optimize their placement, striking a balance between system efficiency, cost effectiveness, and environmental sustainability;
- Instead of real circumstances, we utilize an ad hoc server for simulations with various scenarios and parameters. Then, the performances of the proposed algorithms are analyzed for obtained outcomes through those simulations using various settings and scenarios; as well, detailed discussions are provided for the obtained results.

In the following sections of this paper, we have systematically arranged our discussion and analysis to offer a comprehensive perspective on our research. In Section 2, we provide a detailed problem definition and present an overview of the system. This section serves as the foundation of our study, outlining the specific challenges associated with communication in smart buildings and the system parameters that our proposed algorithms operate within. In Section 3, we introduce our two algorithms for the preemptive placement of nodes within thin walls and the adjacent spaces. Each algorithm is explained in detail, including its design principles, operation, and expected performance characteristics. Our objective here is to provide a thorough understanding of the mechanisms of these algorithms and how they aim to solve the problem defined in Section 2. In Section 4, we delve into an evaluation of our proposed algorithms. Using a series of simulations, we illustrate the performance of each algorithm under various conditions. This section provides explanations of how our algorithms function, demonstrating their potential to improve communication within smart buildings. In Section 5, we conduct a comparative study of the two algorithms. Drawing on the results from the previous section, we analyze and contrast the performance of each algorithm. This comparative analysis allows us to identify the relative strengths and weaknesses of each algorithm, offering valuable insights into which one provides a more optimal solution to the problem of communication disruption and secure surveillance with energy efficiency in smart buildings.

2. Proposed Framework

First of all, we design the efficient barrier model for solidifying smart building surveillance in harsh environments with walls and structures. And the essential terms and definitions in regard to the proposed model are represented. Also, the primary research problem in the paper is formally defined.

2.1. System Overview and Assumptions

The proposed system revolves around a smart building environment, considered as a three-dimensional space, wherein certain areas are obstructed by thin walls that act as physical barriers for communication. These walls divide the space into two parts, creating a challenge for data transfer between different sections of the building. The system members, or sensors, are randomly deployed throughout the available space, excluding the wall. These sensors are the key components in our communication system, serving as the nodes that facilitate data transfer across the building. Their placement is random, reflecting the unpredictability and variation found in real-world deployment. The wall in this system, though physically thin, is considered impenetrable for the communication signals used by the sensors.

Figure 1 depicts a brief overview of the given space. When we consider a two-dimensional plane, a thin wall is located vertically, which may affect surveillance and communication between the left border and right border.

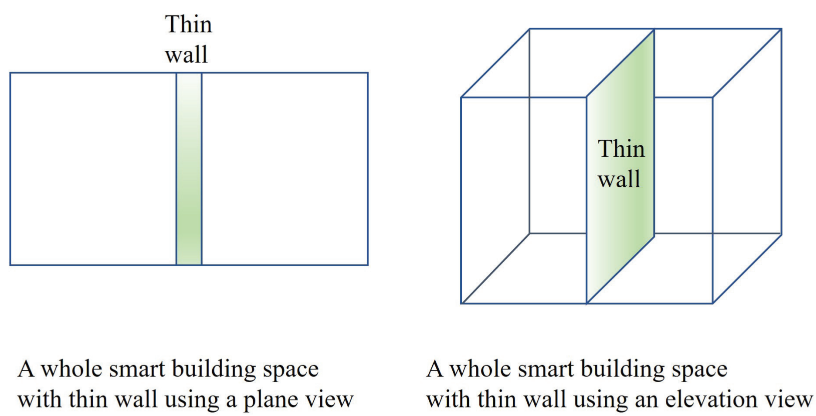


Figure 1. A brief overview of the whole space.

Then, the below assumptions and settings are engraved to activate the proposed system:

- The three-dimensional space is considered as the region of interest within the smart building as a whole. And the smart building contains thin walls, which are located everywhere within the building;
- The proposed system consists of a group of system members or components, including IoT devices, mobile robots, and sensors, where each component has equal detection or communication range and is equipped with wireless transmitters and receivers;
- The connection between two system members is created if there exists an overlapped space between the detection ranges of two neighbors.

2.2. Notations, Essential Terms, Problem Definition

The basic terms which are utilized in the proposed system are presented and the main research problem is also defined in this subsection. The goal is to create a barrier in a three-dimensional space, reducing the number of nodes. There is a very thin wall in the space that separates the two spaces. The input is the sensing radius and the output returns the number of nodes used when the number of barriers is greater than or equal to a certain number of barriers.

Definition 1 (*wall-recognition surveillance security barriers*)**.** *Suppose that there is a smart building space, where the space includes walls or similar complex infrastructure that may affect wireless communication, data transfer, transmission, and reflection. Given the space with thin walls, the system allows heterogeneous members, including a group of IoT devices, mobile robots, sensors, and cameras, which are equipped with a wireless transmitter and receiver. And each member has the maximum allowed number of connections through the thin wall that covers one hop distance or detection range of the system member. The wall-recognition surveillance security barriers in smart buildings, called WalRecogSurv, are constructed by a sub-group of system members to detect any penetration or object movement between specific directions.*

Definition 2 (*MinWalRecogSurv problem*)**.** *It is given that it is necessary to generate a group of wall-recognition surveillance security barriers in a smart building. The MinWalRecogSurv problem is to minimize the total number of wall-recognition surveillance security barriers in the smart building environment, such that the requested allowed number of connections through walls or installations within the walls is satisfied.*

Hence, the objective of the *MinWalRecogSurv* problem is to

$$\textbf{Minimize}\ \delta \tag{1}$$

Also, the indispensable notations, with their brief descriptions and explanations, are summarized in Table 1.

Table 1. Notations.

Notations	Descriptions
S	a 3D smart building surveillance space
M	a set of system members
W	a set of wall-recognition surveillance security barriers
δ	the number of system members
r	the detection range of system member
t	the allowed number of connections through the wall
p	the possible number of connections among system members
q	the requested number of wall-recognition surveillance security barriers
i	an identifier of a system member, where $i \leq \delta, m_i \in M$
j	an identifier of a system member, where $j \leq \delta, m_j \in M$
k	an identifier of a wall-recognition barrier, where $k \leq q, w_k \in W$

3. Proposed Methods

This section presents our proposed schemes to resolve the *MinWalRecogSurv* problem in the smart building space. The implementation processes and steps of both approaches are specified.

3.1. Algorithm 1: Centralized Node Deployment

First of all, an approach for centralizing node position, referred to as *Centralized Node Deployment*, is devised to solve the *MinWalRecogSurv* problem, which returns δ as the minimal number of system members required to build wall-recognition surveillance security barriers. The *Centralized Node Deployment* scheme largely consists of the following steps:

- The first step is to place the nodes in a row along the centerline of the thin wall. This centralized deployment ensures that nodes are evenly distributed along the length of the wall, which is important for maintaining consistent communication coverage;
- When nodes are placed inside thin walls, the algorithm randomly deploys nodes on both sides of adjacent walls. Randomness here means that nodes are placed at various points on adjacent walls, but within a defined range, to ensure effective signal transmission with nodes within thin walls. This step introduces a variation factor that reflects real-world conditions, in which nodes can be placed in various locations, depending on the specific requirements and constraints of the building;
- The final step is to form a communication barrier based on nodes placed inside the thin wall by [35]. This barrier overcomes communication interruptions caused by thin walls and enables effective data transfer between randomly placed nodes on either side of the wall. The formation of this communication barrier optimizes communication paths between nodes and improves data transmission within smart buildings. Then, we estimate the total number of current surveillance barriers and return it as the final outcome.

Figure 2 shows the implementation strategy of Algorithm 1: *Centralized Node Deployment*, with consideration of a centerline in the wall. As can be seen in Figure 2, such a strategy ensures that system members or nodes are distributed evenly in the given smart building space. Also, Figure 3 depicts the executed state of Algorithm 1: *Centralized Node Deployment*. As shown in Figure 3, Algorithm 1 helps the fair distribution of system members through the wall when the wall-recognition surveillance security barriers in the smart building space are created with the requested number of allowed connections through the wall, or installations within the wall, consequently.

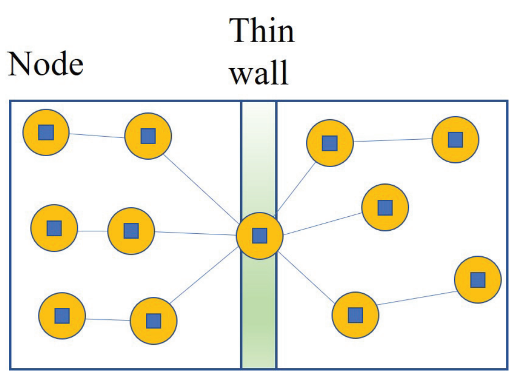


Figure 2. The implementation procedure of Algorithm 1 along the centerline of the wall.

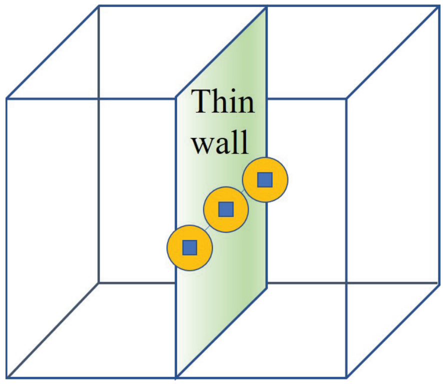


Figure 3. The execution status of Algorithm 1 with the determined node deployments within the wall.

Algorithm 1 Centralized Node Deployment
Inputs: S, M, r, t, q, Output: δ

```
verify M with r within S;
recognize the walls in S;
set W ← ∅;
place nodes in centerline in the walls;
while q number of WalRecogSurv are not formed do
    seek a new WalRecogSurv through the centerline in the walls with t and p;
    if a new WalRecogSurv w_k is found then
        set W ← W ∪ w_k;
    end if
end while
calculate |W|;
update |W| to δ;
return δ;
```

Furthermore, the pseudocode of *Centralized Node Deployment* is explained in Algorithm 1 with formal representations.

3.2. Algorithm 2: Adaptation Node Deployment

Secondly, an approach for adapting node position, called *Adaptation Node Deployment*, is developed to work out the defined *MinWalRecogSurv* problem, seeking the minimal number of system members such that the requested number of allowed connections through the wall, or installations within the wall, is met. Then, the *Adaptation Node Deployment* approach is largely composed of the procedures below:

- The first step assumes that there is no wall and randomly deploys nodes in the entire space of the smart building. This random placement reflects the variability in and irregularity of node placement in the real world;
- This step creates a barrier based on the initial node placement by [35]. This barrier assumes that there is no wall and forms a communication flow between nodes; each node can transmit and receive data to and from adjacent nodes. After the barrier is created, it finds this flow to see how communication is formed;
- After finding the flow, it finds the point where the flow and the wall intersect. This intersection is an area where communication disconnection may occur, and additional nodes are placed at that point to resolve this. This keeps the communication flow through the wall smooth and enables data transfer to other areas within the smart building. Then, we measure the total number of current surveillance barriers and return it as final result.

Figure 4 presents the arbitrary deployment of Algorithm 2: *Adaptation Node Deployment*. The random deployments are performed through the entire space of the smart building. Then, Figure 5 describes the implementation procedure of Algorithm 2. As can be seen in Figure 5, Algorithm 2 searches for the flow to see how communication is formed after the barriers are generated. Moreover, Figure 6 shows the execution status of Algorithm 2 with the node locations adopted within the wall. It follows that, after finding the flows, Algorithm 2 recognizes the wall intersections so that it keeps the communication flow and the detection through the wall within the smart building.

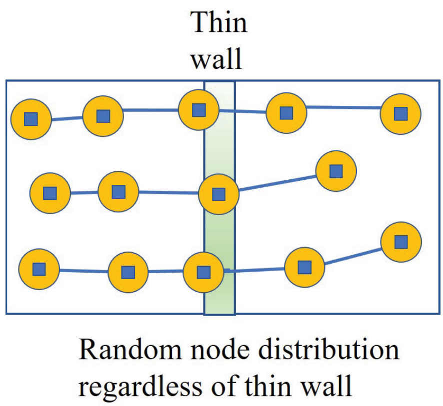


Figure 4. The arbitrary deployment of Algorithm 2.

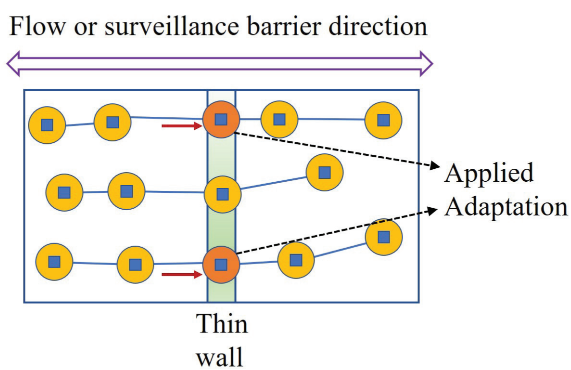


Figure 5. The implementation procedure of Algorithm 2 with consideration of adapted flows.

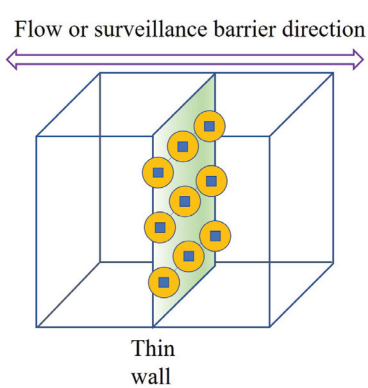


Figure 6. The execution status of Algorithm 2 with the node locations adopted within the wall.

Algorithm 2 Adaptation Node Deployment
Inputs: S, M, r, t, q, Output: δ

identify M with r within S;
detect the walls in S;
set $W \leftarrow \emptyset$;
while q number of flows are not generated **do**
    seek a new flow between left border and right border with t and p;
    if a new flow is found **then**
        add it to W;
    end if
end while
calculate $|W|$;
search for the points where the flow and the wall intersect;
add those points to $|W|$;
update $|W|$ to δ;
return δ;

Moreover, the pseudocode of *Adaptation Node Deployment* is specified in Algorithm 2, based on formal notations and descriptions.

4. Performance Analysis

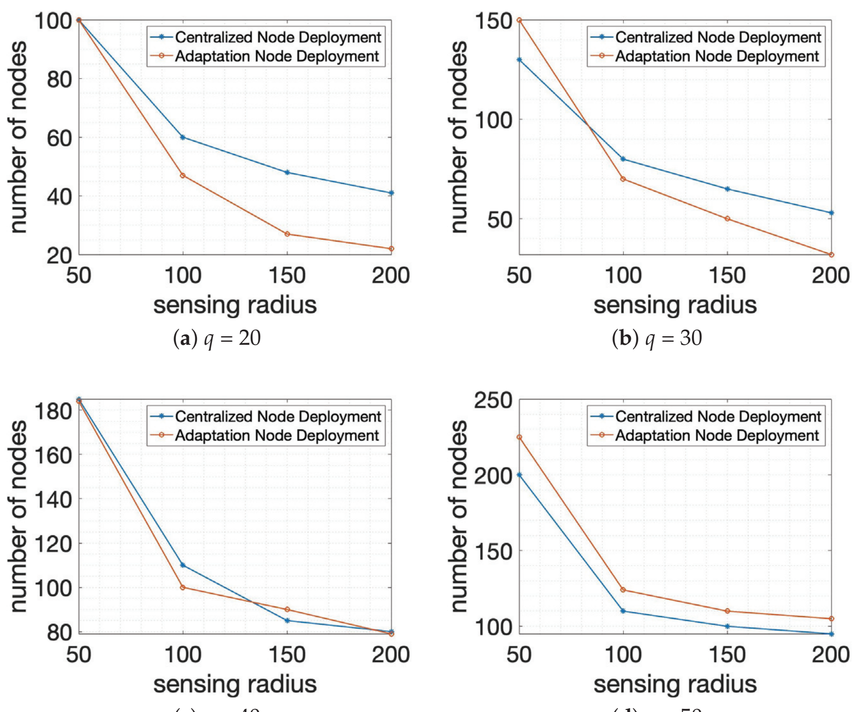
In this section, we evaluate the performances of the proposed Algorithm 1: *Centralized Node Deployment* and Algorithm 2: *Adaptation Node Deployment* after several groups of simulations are performed. In the simulations, we utilize several settings and parameters, covering different sensing or detection ranges of system members, different numbers of connections through the wall, different possible numbers of connections among system members, several numbers of wall-recognition surveillance security barriers, etc. The simulation settings are summarized as follows. The size of the smart building is considered as a 1000 (width) $\times$ 1000 (height) $\times$ 1000 (depth) space. The sensing or detection range of system member r ranges from 50 to 200, where each system member has equal detection radius. In essence, a sole thin wall in the smart building is considered in each simulation. The allowed number of connections through the wall t ranges from 10 to 30. And the possible number of connections among system members p is considered between 1 and 4. Also, the requested number of wall-recognition surveillance security barriers ranges from 20 through 50. As such, the objective value of δ is the number of system members, which is the final output value of the proposed algorithms and the average value of 100 different graphs and experiments. All experiments are conducted using C^{++} in an arm64cpu computer; the resulting graphs are created by MATLAB.

First, our schemes are described in Table 2, including the pros and cons compared to other studies.

Table 2. Pros and sons of previous studies and our scheme.

Studies	Pros	Cons
[23]	- Initial work of barriers - Sleep-wakeup scheduling - Homogeneous capability - Heterogeneous capability	- 2D environment - Not practical product - Biased theoretical analysis - Not expanded environment
[25]	- Controllable trajectories - Static and mobile sensors - Bidding mechanism - Deterministic countermeasures	- 2D environment - Not practical product - Biased theoretical analysis - Not expanded environment
[33]	- Two-way-enabled barriers - Slab dividing strategy - Perpendicular detection - Horizontal detection	- 2D environment - Not practical product - Biased simulation analysis - Not expanded environment
Our scheme	- 3D environment - Smart building with thin wall - Green property - Deployment strategy with wall	- Sole thin wall - Not practical product - Biased simulation analysis - Not expanded environment

In the first group of experiments, Algorithm 1: *Centralized Node Deployment* and Algorithm 2: *Adaptation Node Deployment* are performed over different sensing ranges with the allowed number of connections through the wall $t = 20$ and $p = 3$ in the $1000 \times 1000 \times 1000$ smart building size, as shown in Figure 7. It is noted that the experimental outcome is composed of two axes, so that the X-coordinate specifies the sensing range of the system members and the Y-coordinate presents the total number of system members δ of objective value, so as to build the requested number of wall-recognition surveillance security barriers completely. In Figure 7, sensing radius or detection range has been set as 50, 100, 150, or 200. Figure 7a,b demonstrates the performance of two different algorithms according to different sensing ranges with $q = 20$ and $q = 30$, respectively. Also, Figure 7c,d shows the performance comparison of two algorithms when $q = 40$ and $q = 50$ are given in the experiment. As shown in Figure 7, it is verified that the total number of system members δ is decreasing as the sensing range of node is increasing because the bigger sensing range allows more space to be detected by each node. Also, we can confirm that Algorithm 2: *Adaptation Node Deployment* shows better performance than Algorithm 1: *Centralized Node Deployment* in the first experimental scenario.

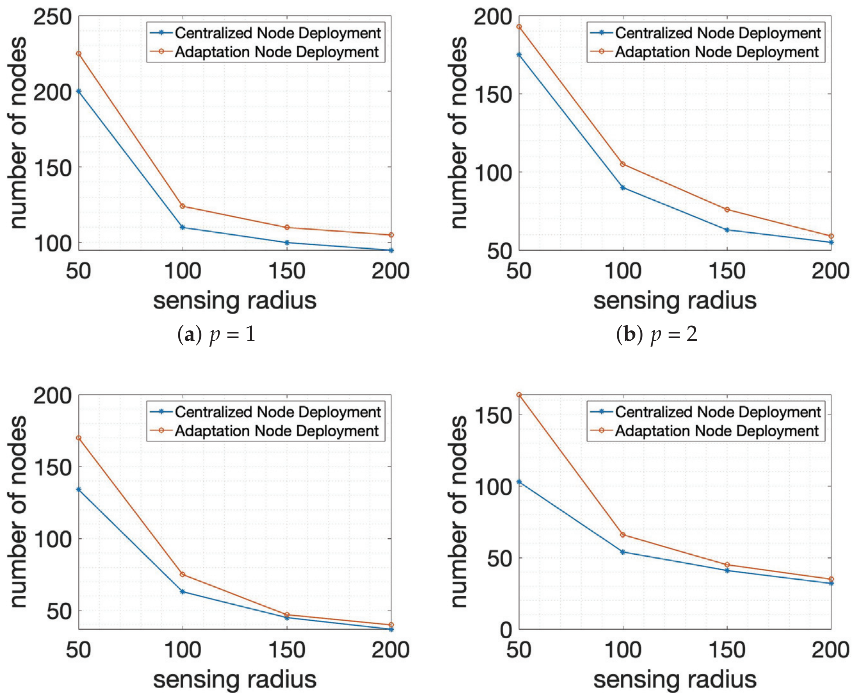


(**a**) $q = 20$ (**b**) $q = 30$

(**c**) $q = 40$ (**d**) $q = 50$

Figure 7. Performance comparison for the total number of nodes or system members of the requested number of wall-recognition surveillance security barriers q in different sensing ranges with the allowed number of connections through the wall $t = 20$ and $p = 3$ in $1000 \times 1000 \times 1000$ smart building size.

For the second set of simulations, we also executed two algorithms, Algorithm 1: *Centralized Node Deployment* and Algorithm 2: *Adaptation Node Deployment*, using various sensing radii with the allowed number of connections through the wall $t = 20$ and $q = 50$ in $1000 \times 1000 \times 1000$ smart building size, as can be seen in Figure 8. Similar to the first group of experiments, the experimental outcome results consist of two axes, where the X-coordinate represents the sensing radius of system members and the Y-coordinate specifies the total number of system members δ as the final outcome. In Figure 8, sensing radius or detection range has been set as 50, 100, 150, or 200. Figure 8a,b shows the performance comparison if two algorithms are executed with $p = 1$ and $p = 2$. And Figure 8c,d stands for the performance of two algorithms when $p = 3$ and $p = 4$ are utilized in the system. According to Figure 8, it is observed that the total number of system members δ is decreasing significantly as the sensing range of the node is increasing. The reason is that the larger sensing range gives more opportunity to search for neighbors or system members when the wall-recognition surveillance security barriers are formed. Moreover, it is demonstrated that Algorithm 2: *Adaptation Node Deployment* outperforms Algorithm 1: *Centralized Node Deployment* for all applicable missions in the second scenario of simulation. And the performance difference between two algorithms is diminished if the sensing range of system members increases.



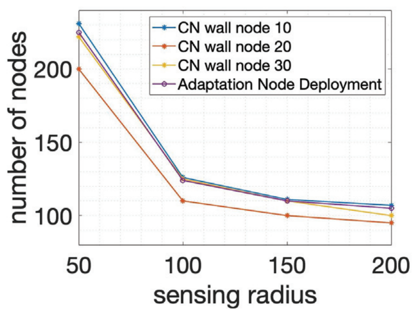
(**a**) $p = 1$ (**b**) $p = 2$

(**c**) $p = 3$ (**d**) $p = 4$

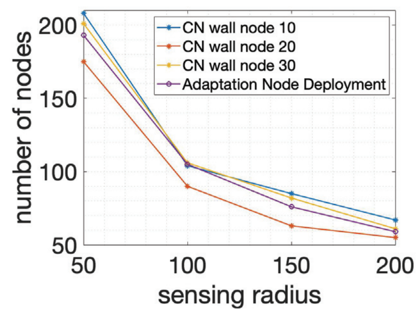
Figure 8. Performance comparison for the total number of nodes or system members of the possible number of connections among system members in different sensing ranges with the allowed number of connections through the wall $t = 20$ and $q = 50$ in $1000 \times 1000 \times 1000$ smart building size.

Lastly, as the third group of experiments, we achieved two schemes for Algorithm 1: *Centralized Node Deployment* and Algorithm 2: *Adaptation Node Deployment* based on the scenario that covers the requested number of wall-recognition surveillance security barriers q in different sensing ranges with the allowed number of connections through the wall $t = 10, 20, 30$ and $p = 3$ in $1000 \times 1000 \times 1000$ smart building size. In particular, Algorithm 1: *Centralized Node Deployment* with various t values for the allowed number of connections through the wall is implemented and is compared with Algorithm 2: *Adaptation Node Deployment*. Similar to previous groups of experiments, the simulation results are presented with two axes, in which the X-coordinate stands for the sensing radius of system members and the Y-coordinate represents the total number of system members δ for the obtained result. Figure 9a,b depicts the performance comparison for Algorithm 1: *Centralized Node*

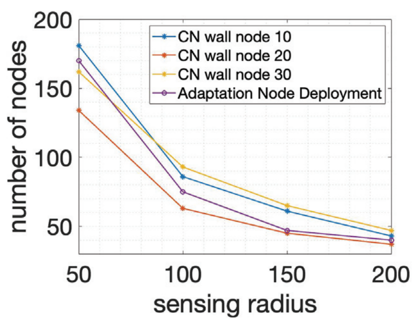
Deployment with the allowed number of connections through the wall t = 10, 20, 30 and Algorithm 2: *Adaptation Node Deployment*, depending on q = 50 and q = 40. And Figure 9c,d presents the results of Algorithm 1: *Centralized Node Deployment* with t = 10, 20, 30 and Algorithm 2: *Adaptation Node Deployment* when the requested number of wall-recognition surveillance security barriers q = 30 and q = 20 are given. From Figure 9, it is identified that the total number of system members δ is decreasing significantly as the sensing range of the node is increasing as a whole because the larger sensing range of each node gives a higher chance to cover a wide space and to connect with other nodes. In addition, Algorithm 1: *Centralized Node Deployment* with t = 20 has the best performance compared to the other cases in the third scenario of simulation.



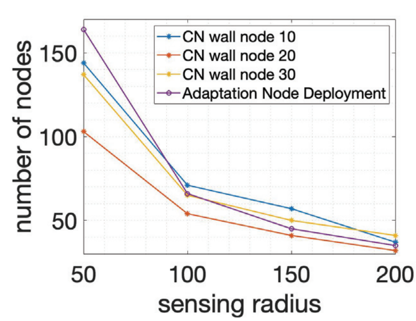
(**a**) q = 50



(**b**) q = 40



(**c**) q = 30



(**d**) q = 20

Figure 9. Performance comparison for the total number of nodes or system members of the requested number of wall-recognition surveillance security barriers q in different sensing ranges with the allowed number of connections through the wall t = 10, 20, 30 and p = 3 in 1000 × 1000 × 1000 smart building size.

5. Conclusions

In this paper, we proposed and evaluated two distinct algorithms, *Centralized Node Deployment* and *Adaptation Node Deployment*, to overcome the challenge of communication disruption in smart buildings caused by physical barriers like thin walls. Our findings underscore the effectiveness of both algorithms, with their unique deployment strategies contributing to the optimal functioning of the communication system within the building. The *Centralized Node Deployment* algorithm, with its strategic node placement along the thin walls, proved effective in maintaining consistent communication coverage and effectively mitigating potential communication disruptions. Notably, it showed superior performance as the number of required barriers increased, indicating its ability to handle complex communication obstacles. On the other hand, the *Adaptation Node Deployment* algorithm, starting with random node placement and adapting over time, also demonstrated its capability to ensure efficient communication across the building. While its initial performance varied, over extended periods, its performance converged with that of the *Centralized Node Deployment* algorithm. Interestingly, as the lifetime of the system increased, the performance gap between the two algorithms diminished. This finding indicates that both algorithms, despite their differing initial strategies, are well-suited for long-term communication optimization in smart buildings. Overall, our study contributes valuable insights into the strategic placement of communication nodes in smart buildings, aiming to facilitate seamless and efficient communication in the face of physical barriers. We believe that our research will serve as a strong foundation for future work in this area, potentially leading to even more efficient algorithms and strategies for communication in smart buildings. Moreover, we plan to expand smart complex infrastructure consisting of multiple number of thin walls and thick walls, as well as to apply realistic experimental environments based on the proposed framework and strategies.

Author Contributions: Conceptualization, T.L.; software, T.L.; validation, T.L.; investigation, T.L. and H.K.; methodology, T.L.; resources, T.L.; data curation, T.L.; writing—draft preparation, T.L. and H.K.; writing—review and editing, H.K.; visualization, T.L. and H.K.; supervision, H.K.; project administration, H.K.; funding acquisition, H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
UAVs	Unmanned Aerial Vehicles
WalRecogSurv	wall-recognition surveillance security barriers in smart buildings

References

1. Sha, Q.; Liu, X.; Ansari, N. Efficient Multiple Green Energy Base Stations Far-Field Wireless Charging for Mobile IoT Devices. *IEEE Internet Things J.* **2023**, *10*, 8734–8743. [CrossRef]
2. Bugshan, N.; Khalil, I.; Rahman, M.S.; Atiquzzaman, M.; Yi, X.; Badsha, S. Toward trustworthy and privacy-preserving federated deep learning service framework for industrial internet of things. *IEEE Trans. Ind. Inform.* **2023**, *19*, 1535–1547. [CrossRef]
3. Taha, A.M.; Elabd, A. IoT for certified sustainability in smart buildings. *IEEE Netw.* **2023**, *35*, 241–247. [CrossRef]
4. Ko, H.; Pack, S.; Leung, V.C.M. Performance optimization of serverless computing for latency-guaranteed and energy-efficient task offloading in energy-harvesting industrial IoT. *IEEE Internet Things J.* **2023**, *10*, 1897–1907.
5. Baek, H.; Ko, H.; Kim, J.; Jeon, Y.; Pack, S. Sensing quality-aware task allocation for multidimensional vehicular urban sensing. *IEEE Internet Things J.* **2023**, *10*, 9989–9998. [CrossRef]
6. Zhang, Q.; Wu, J.; Zanella, M.; Yang, W.; Bashir, A.K.; Fornaciari, W. Sema-IIoVT: Emergent semantic-based trustworthy information-centric fog system and testbed for intelligent internet of vehicles. *IEEE Consum. Electron. Mag.* **2023**, *12*, 70–79. [CrossRef]
7. Filali, A.; Mlika, Z.; Cherkaoui, S.; Kobbane, A. Dynamic SDN-based radio access network slicing with deep reinforcement learning for URLLC and eMBB Services. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 2174–2187. [CrossRef]
8. Dao, N.; Kim, Y.; Jeong, S.; Park, M.; Cho, S. Achievable multi-security levels for lightweight IoT-enabled devices in infrastructureless peer-aware communications. *IEEE Access* **2017**, *5*, 26743–26753. [CrossRef]
9. Xu, S.; Liu, J.; Kato, N.; Du, Y. Intelligent reflecting surface backscatter enabled multi-tier computing for 6G internet of things. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 320–333. [CrossRef]
10. Su, Y.; Gao, Z.; Du, X.; Guizani, M. User-centric base station clustering and resource allocation for cell-edge users in 6G ultra-dense networks. *Future Gener. Comput. Syst.* **2023**, *141*, 173–185. [CrossRef]

11. Kim, J.; Lee, J.; Ko, H.; Kim, T.; Pack, S. Space mobile networks: Satellite as core and access networks for B5G. *IEEE Commun. Mag.* **2022**, *60*, 58–64. [CrossRef]
12. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Niyato, D.; Dobre, O.A.; Poor, H.V. 6G Internet of Things: A comprehensive survey. *IEEE Internet Things J.* **2022**, *9*, 359–383. [CrossRef]
13. Kovtun, V.; Izonin, I.; Gregus, M. Mathematical models of the information interaction process in 5G-IoT ecosystem: Different functional scenarios. *ICT Express* **2023**, *9*, 264–269. [CrossRef]
14. Geraci, G.; García-Rodríguez, A.; Azari, M.M.; Lozano, A.; Mezzavilla, M.; Chatzinotas, S.; Chen, Y.; Rangan, S.; Renzo, M.D. What will the future of UAV cellular communications be? A flight from 5G to 6G. *IEEE Commun. Surv. Tutorials* **2022**, *24*, 1304–1335. [CrossRef]
15. Dao, N.; Park, M.; Kim, J.; Paek, J.; Cho, S. Resource-aware relay selection for inter-cell interference avoidance in 5G heterogeneous network for Internet of Things systems. *IEEE Commun. Surv. Tutorials* **2019**, *93*, 877–887. [CrossRef]
16. Yao, J.; Ansari, N. QoS-Aware Machine Learning Task Offloading and Power Control in Internet of Drones. *IEEE Internet Things J.* **2023**, *10*, 6100–6110. [CrossRef]
17. Raja, G.; Senthivel, S.G.; Balaganesh, S.; Rajakumar, B.R.; Ravichandran, V.; Guizani, M. MLB-IoD: Multi Layered Blockchain Assisted 6G Internet of Drones Ecosystem. *IEEE Trans. Veh. Technol.* **2023**, *72*, 2511–2520. [CrossRef]
18. Liu, L.; Xiong, K.; Cao, J.; Lu, Y.; Fan, P. Letaief, K.B. Average AoI minimization in UAV-assisted data collection with RF wireless power transfer: A deep reinforcement learning scheme. *IEEE Internet Things J.* **2022**, *9*, 5216–5228. [CrossRef]
19. Cong, J.; Wang, X.; Yan, C.; Yang, L.T.; Dong, M.; Ota, K. CRB Weighted Source Localization Method Based on Deep Neural Networks in Multi-UAV Network. *IEEE Internet Things J.* **2023**, *10*, 5747–5759. [CrossRef]
20. Kim, H.; Ben-Othman, J.; Mokdad, L.; Son, J.; Li, C. Research challenges and security threats to AI-driven 5G virtual emotion applications using autonomous vehicles, drones, and smart devices. *IEEE Netw.* **2020**, *34*, 288–294. [CrossRef]
21. Memos, V.A.; Psannis, K.E. Optimized UAV-based data collection from MWSNs. *ICT Express* **2023**, *9*, 29–33. [CrossRef]
22. Hendria, W.F.; Phan, Q.T.; Adzaka, F.; Jeong, C. Combining transformer and CNN for object detection in UAV imagery. *ICT Express* **2023**, *9*, 258–263. [CrossRef]
23. Kumar, S.; Lai, T.H.; Posner, M.E.; Sinha, P. Maximizing the lifetime of a barrier of wireless sensors. *IEEE Trans. Mob. Comput.* **2010**, *9*, 1161–1172. [CrossRef]
24. Carrabs, F.; Cerulli, R.; D'Ambrosio, C.; Raiconi, A. A hybrid exact approach for maximizing lifetime in sensor networks with complete and partial coverage constraints. *J. Netw. Comput. Appl.* **2015**, *58*, 12–22. [CrossRef]
25. Vecchio, M.; Lopez-Valcarce, R. Improving area coverage of wireless sensor networks via controllable mobile nodes: A greedy approach. *J. Netw. Comput. Appl.* **2015**, *48*, 1–13. [CrossRef]
26. Benahmed, T.; Benahmedm, K. Optimal barrier coverage for critical area surveillance using wireless sensor networks. *Int. J. Commun. Syst.* **2019**, *32*, 1–21. [CrossRef]
27. Daniel, A.M.; Mirshak, R. Fault-tolerant design of barrier coverage for periodically repairable wireless sensor networks. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 5802–5822. [CrossRef]
28. Chen, G.; Xiong, Y.; She, J. A k-barrier coverage enhancing scheme based on gaps repairing in visual sensor network. *IEEE Sens. J.* **2023**, *23*, 2865–2877. [CrossRef]
29. Chang, J.; Shen, X.; Bai, W.; Li, W. Energy-efficient barrier coverage based on nodes alliance for intrusion detection in underwater sensor networks. *IEEE Sens. J.* **2022**, *22*, 3766–3776. [CrossRef]
30. Wang, Z.; Chen, H.; Cao, Q.; Qi, H.; Wang, Z.; Wang, Q. Achieving location error tolerant barrier coverage for wireless sensor networks. *Comput. Netw.* **2017**, *112*, 314–328. [CrossRef]
31. Kim, H.; Ben-Othman, J. Eco-friendly low resource security surveillance framework toward green AI digital twin. *IEEE Commun. Lett.* **2023**, *27*, 377–380. [CrossRef]
32. Si, P.; Fu, Z.; Shu, L.; Yang, Y.; Huang, K.; Liu, Y. Target-barrier coverage improvement in an insecticidal lamps internet of UAVs. *IEEE Trans. Veh. Technol.* 2022, *71*, 4373–4382. [CrossRef]
33. Kim, H.; Ben-Othman, J.; Mokdad, L.; Bellavista, P. A virtual emotion detection architecture with two-way enabled delay bound toward evolutional emotion-based IoT services. *IEEE Trans. Mob. Comput.* **2022**, *21*, 1172–1181. [CrossRef]
34. Li, W.; Zhang, W. Coverage hole and boundary nodes detection in wireless sensor networks. *J. Netw. Comput. Appl.* **2015**, *48*, 819–827. [CrossRef]
35. Lammich, P.; Sefidgar, S.R. Formalizing the Edmonds-Karp algorithm. In *Springer Lecture Notes in Computer Science (LNCS)*; Springer: Berlin/Heidelberg, Germany, 2016.

sensors

MDPI

Article

The Development of a Stereo Vision System to Study the Nutation Movement of Climbing Plants

Diego Rubén Ruiz-Melero [1], Aditya Ponkshe [2], Paco Calvo [2,3] and Ginés García-Mateos [1,*]

[1] Computer Science and Systems Department, University of Murcia, 30100 Murcia, Spain; diegoruben.ruiz@um.es
[2] Minimal Intelligence Laboratory (MINT Lab), University of Murcia, 30100 Murcia, Spain; ponkshe.aditya@gmail.com (A.P.); fjcalvo@um.es (P.C.)
[3] Department of Philosophy, University of Murcia, 30100 Murcia, Spain
* Correspondence: ginesgm@um.es

Abstract: Climbing plants, such as common beans (*Phaseolus vulgaris* L.), exhibit complex motion patterns that have long captivated researchers. In this study, we introduce a stereo vision machine system for the in-depth analysis of the movement of climbing plants, using image processing and computer vision. Our approach involves two synchronized cameras, one lateral to the plant and the other overhead, enabling the simultaneous 2D position tracking of the plant tip. These data are then leveraged to reconstruct the 3D position of the tip. Furthermore, we investigate the impact of external factors, particularly the presence of support structures, on plant movement dynamics. The proposed method is able to extract the position of the tip in 86–98% of cases, achieving an average reprojection error below 4 px, which means an approximate error in the 3D localization of about 0.5 cm. Our method makes it possible to analyze how the plant nutation responds to its environment, offering insights into the interplay between climbing plants and their surroundings.

Keywords: plant nutation movement; computer vision; image processing; climbing plants

Citation: Ruiz-Melero, D.R.; Ponkshe, A.; Calvo, P.; García-Mateos, G. The Development of a Stereo Vision System to Study the Nutation Movement of Climbing Plants. *Sensors* **2024**, *24*, 747. https://doi.org/10.3390/s24030747

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 6 December 2023
Revised: 21 January 2024
Accepted: 22 January 2024
Published: 24 January 2024

1. Introduction

The nutational movement of climbing plants refers to the rhythmic, circular, or nodding motion exhibited by certain plant parts, such as stems, tendrils, or growing tips, as they explore and interact with their environment during the process of climbing [1]. This movement is often associated with the search for support structures, such as poles, trellises, or other plants, which the climbing plant can use for stability and upward growth [2]. It occurs as a result of differential growth rates on different sides of the plant organ. As the plant grows, the cells on one side of the organ elongate more rapidly than those on the opposite side, causing the organ to bend or curve [3]. This bending or curving allows the plant to explore its surroundings and find a suitable support for climbing.

Climbing plants, such as common beans (*Phaseolus vulgaris* L.), employ various mechanisms for climbing, including twining, where the plant winds around a support, and the use of tendrils, specialized structures that coil around objects for support [4].

The study of nutation movements in climbing plants is part of the broader field of plant tropisms, which involves the growth responses of plants to external stimuli such as light, gravity, and mechanical touch, first studied in the pioneering works of Charles Darwin [5]. This nutation movement allows climbing plants to optimize their growth in response to their environment, increasing their chances of successful climbing and their access to sunlight for photosynthesis. Moreover, it has been the inspiration for the development of new techniques and methods in robotics [6], computational intelligence [7,8], and other bioinspired innovations [9].

This current study of nutation and other plant movements requires the recording of time-lapse images of plants, due to the long time span in which the movements occur, and the use of artificial vision algorithms for the efficient analysis of the videos obtained. For example, Navarro et al. [10] developed a specific capture chamber to monitor the growth of plants at intervals from 30 s to hours or even days. The captured images were then analyzed with computer vision algorithms based on thresholding, morphological operators, and blobs detection. Using the parameters obtained from the blobs, features such as area, length, compactness, and perimeter were extracted for further analysis. Stolarz et al. [11] developed software called the "Circumnutation Tracker" to analyze the circular movement of the tip of the plants in zenithal time-lapse videos. The X and Y positions of the tip are manually selected by the user, and the system includes useful tools for the visualization and characterization of the plant movement, automatically extracting circumnutation parameters.

More recently, Tenzer and Clifford [12] proposed a technique to analyze the growing movement of hydroponic plants using different neural networks on grayscale images. U-net, Linknet, FPN, PSPNet, and a 34-layer ResNet architecture were used for comparison, obtaining a maximum area under the curve (AUC) for the validation set over 0.92. Another recent method based on deep learning was presented by Mao et al. [13], which proposed a U-net network architecture to segment the plant. This method is an improvement of their free software "Plant Tracer", which analyzed plant movement using the classic computer vision techniques of object tracking and blob detection. Díaz-Galián et al. [14] presented a methodology to analyze the movement of plants using different reference points, which were obtained from image analysis. This methodology included coordinates transformation, curve fitting, and statistical analysis, allowing a comparison between different plant species. Plant movement has also been analyzed using other types of sensors. For example, Geldhof et al. [15] developed a digital inertial measurement unit (IMU) sensor to measure in real-time the movement of leaves, achieving a precision of pitch and roll angles under 0.5°. However, the exact 3D position of the leaves was not obtained.

The aim of this work is to develop a stereo system for the visual tracking of plant nutation movement, which enables obtaining the precise 3D location of the tip of the plant. This work was focused on the study of the common bean (*Phaseolus vulgaris* L.), since the ultimate purpose is to analyze the differences in the movement between using or not using a support pole for plants. Previous work from the MINT Lab research group suggests that the dynamic patterns of plant nutation are influenced by the presence of support to climb in their vicinity [2]. This fact has also been observed by other researchers. For example, Guerra et al. [16] analyzed the 3D kinematics of the movement of climbing plants, demonstrating that plants are not only able to perceive their environment but can also scale the movement of their tendrils depending on the size of the support.

The objective of this present study is not to analyze these kinematic changes under different environmental conditions but to develop computational tools for the accurate 3D measurement of plant tip movement that will be used in further studies. Consequently, for our analysis to be effective in a broader range of conditions, we obtained data on plants from different settings, i.e., with and without a support to climb onto. Since the proposed method is able to reconstruct the 3D position of the tip in both conditions, our method promises to offer a framework for future studies to analyze how plant nutation responds to different environments.

2. Materials and Methods

The proposed computer vision method to track the movement of the plant tip is shown in Figure 1. It consists of five main stages: calibration, data acquisition, plant tip detection in the top and side images, 3D position estimation, and visualization of the results.

In the following subsections, the characteristics of the experimental setup and the stereo imaging system are presented first. Then, a brief introduction to the mathematical foundations and artificial vision algorithms used is given. Finally, the procedure used to estimate the position of the plant tip at each of the moments captured in the time-lapse videos is described.

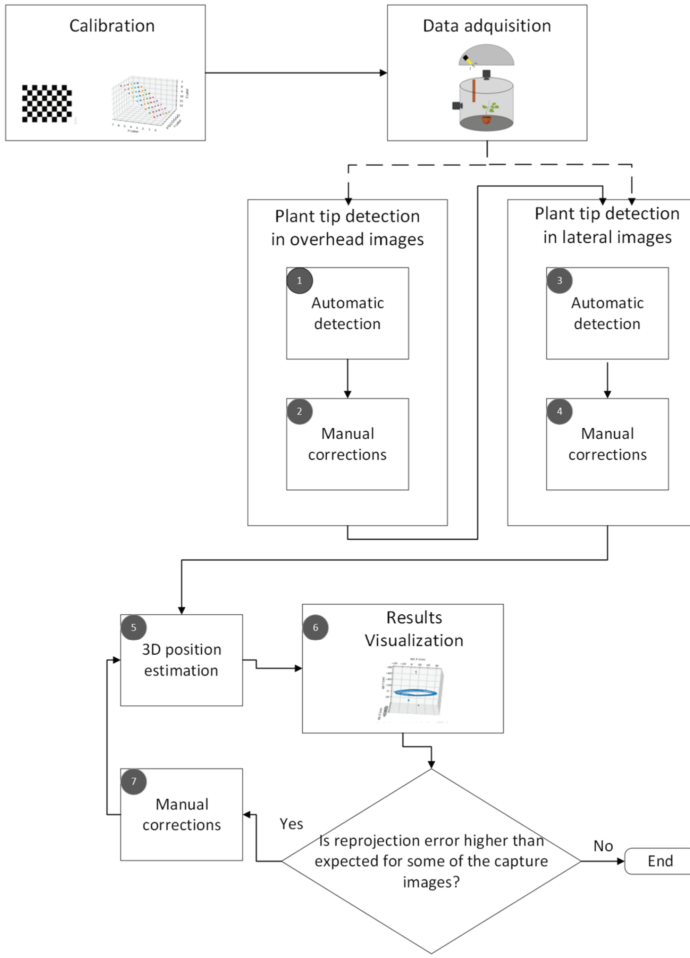


Figure 1. Pipeline for the proposed computer vision method used to track the plant tip movement.

2.1. Experimental Setup

The experiments were carried out by the MINT Lab research group at the Scientific and Technical Research Area of the University of Murcia (Spain). These experiments were conducted under controlled conditions and consisted of two scenarios. In the first scenario, the climbing plant was placed near a pole, which served as a support for the plant to grip when it reached its position; in the second scenario, the plant was placed within a cabin with no pole in it. The pole was 90 cm in height and 1.8 cm in diameter and was placed at a distance of 30 cm from the plant center. Concerning the soil, a mixture of peat moss and perlite (70–30%) was used throughout all the experiments. Both cabins were equipped with white parabolic reflectors to provide symmetrical lighting. The temperature of the growth chamber was kept constant at 20 °C and the relative humidity at 85% ± 5%. We have to note that, although this is a high relative humidity, plants are able to accomplish their life cycles successfully with no harm, especially in controlled conditions in which heat stress can be prevented. A L16:D8 h photoperiod was provided via high-pressure sodium lamps (Lumatek pulse-start HPS Lamp 250 W; height: 150 cm and photon fluence rate: 430 ± 50 µmol m^{-2} s^{-1} at leaf level). During the 8 h of darkness, a dim phototropically inactive green safelight (fluence rate under 5 µmol m^{-2} s^{-1}) was activated, which was enough to take pictures in the dark using a 4.2 μm high dynamic range (115 dB) image sensor.

Figure 2 shows an example of these two scenarios, with and without a supporting pole. Specifically, the plant species used for this experiment were the common bean (*Phaseolus vulgaris* L.). Seeds of the cultivar "Garrafal Oro Vega" variety were provided by Semillas Ramiro Arnedo S.A., Spain (https://www.ramiroarnedo.com). "Garrafal Oro Vega" is a variety with abundant foliage, having medium-sized leaves and containing 18–20 cm long pods. The plants went through multiple stages from germination to the start of the recording. The seeds took 48 h to germinate. Upon germination, the young seedlings were transferred to coconut fiber growing pellets and kept on propagation trays for further development. Once the first true leaves matured, which took around 8–12 days from the young seedling stage, healthy-looking, 25–30 cm tall plants were selected and transferred into the recording booths for video capture. The videos were recorded until the plant either grabbed the pole, or the tip came out of the experimental area (i.e., the area visible to the cameras), which usually happened after 3 or 4 days.

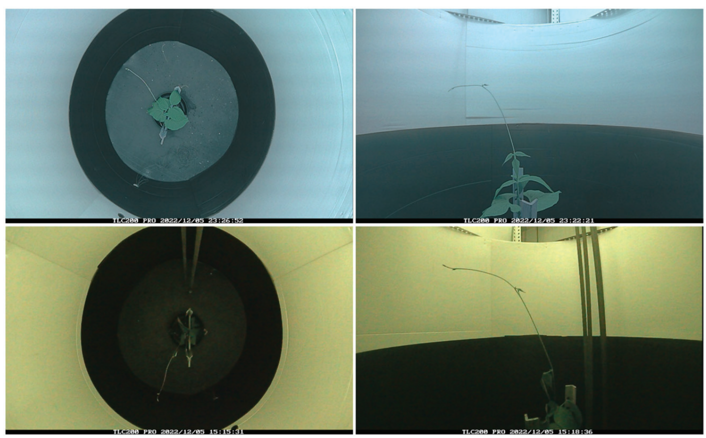


Figure 2. Images of the experiments conducted. At the top, zenithal and lateral images of the experiment in the no-pole condition; at the bottom, images of the pole condition.

It can be observed in Figure 2 that a dark color was used at the bottom part of the booth. The purpose of this dark cover was to increase the contrast with the plant contour, thus enhancing the performance of the algorithm for extracting the position of the plant tip in the videos corresponding to the top view in time-lapse. We made sure that the white part remained disproportionately more than the black part. The proportion of the lower black part was calibrated so that the plants did not show any shade avoidance response, thereby altering the nutation movement.

Once the young seedlings were transferred to the coconut fiber growing pellets and kept on propagation trays for further development, 30 mL of water was initially added every day. Afterwards, when the plants started to mature, they were watered so that around 80% moisture was maintained throughout the growing stage. For recording purposes, the plants were transferred into big, black plastic pots. At the onset of the recording, 400 mL of water was added to each plant, which was found sufficient to keep the soil moist throughout the recording phase. No extra water was added during the four days of recording.

To estimate the nutational movement of the plant, a stereo pair of cameras was used, recording synchronously and in time-lapse. The target of the motion tracking was the plant tip. The configuration of the stereo vision system is shown in Figure 3. This imaging system consisted of two Brinno TLC200 Pro cameras (Brinno Ltd., Taipei, Taiwan) positioned at the top and side of the plant, forming a 90° angle. The technical specifications of the cameras are shown in Table 1. This top-side configuration of the cameras was chosen because it can offer high accuracy in detecting the tip of the plant, although it may not be the best choice for tracking other parts of the plant.

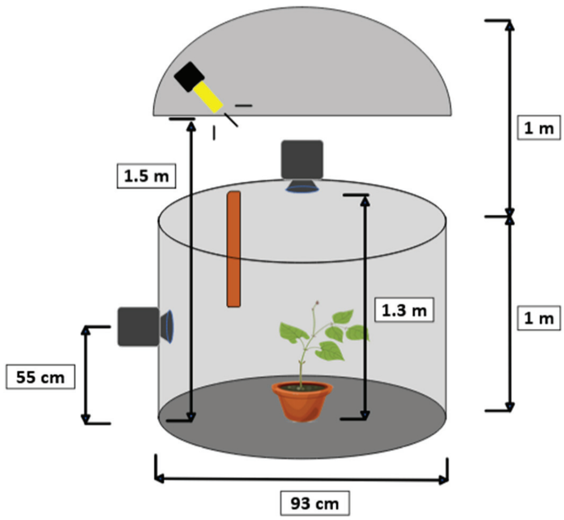


Figure 3. Configuration of the stereo imaging system that was used. The recording cabin was a cylinder of 1 m height and 93 cm radius. The lateral camera was situated at 55 cm from the floor, the zenithal camera at 1.3 m, and the lamp at 1.5 m. The upper reflector had a height of 1 m.

Table 1. Technical specifications of the cameras used in the stereo system.

	Characteristics
Sensor	Type: 1/3″ HDR sensor Dynamic range: 115 db Resolution: 1.3 Mega pixel (1280 × 720 px) Pixel size: 4.2 µm Sensibility: 3650 mV/lux-sec
Optical lens	Type: CS Mount Aperture: F 2.0 Field of view: 112° Focal length: 19 mm

To synchronize the captured images, the recording was started simultaneously on both cameras. Each camera used its own internal clock. The images were captured at one-minute intervals for a continuous 24 h recording, lasting from 3 to 4 days. Because each camera had its own clock, there could be a slight offset in the images between the two cameras for some recordings. To determine the number of frames corresponding to this offset, the light–dark cycle used in the experiments to provide the L16:D8 photoperiod was used. For this purpose, the zenithal images were used as a reference, based on the photoperiod changes. During one of the lighting changes, the offset between the top and side view images was computed, measured as the number of frames of difference. This number was used in the further analysis of the sequence.

2.2. Mathematical Methods

In general, a camera maps 3D points in space to 2D points on the image plane. The cameras used in the experiments were modeled using a pinhole camera model. In a pinhole model, a ray from a point in space passes through a fixed point called the projection center. The intersection of this ray with a chosen plane in space, the image plane, is the projection of the point on the image plane [17]. The projective transformation given by a pinhole camera model is as follows:

$$sp = A[Rt]P_w \tag{1}$$

where:

- s is an arbitrary scaling factor of the projective transformation.
- p is the 2D pixel on the image plane.
- P_w is the 3D point expressed in the world coordinate system.
- A is the matrix with the intrinsic parameters of the camera.
- R and t are the rotation and translation, respectively, describing the coordinate change from the world to the camera coordinate system.

The matrix with the intrinsic parameters is composed of the focal lengths, f_x and f_y, expressed in pixel units, and the principal point of the camera, (C_x, C_y).

$$A = \begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

This matrix with the intrinsic parameters does not depend on the scene, so it can be reused once estimated, as long as the focal length remains fixed. On the other hand, matrix $[R|t]$ contains the extrinsic parameters of the camera. This matrix performs the homogeneous transformation and represents the change of basis from the world coordinate system to the camera coordinate system, as follows:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{3}$$

Combining these two matrices yields the projective transformation, which maps the 3D points in the world to the 2D points on the image plane in normalized camera coordinates.

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{4}$$

The pinhole camera model does not consider the distortion caused by the lenses used in the cameras. To accurately represent a real camera, the camera model includes both radial and tangential lens distortion. Radial distortion occurs when light rays bend more at the edges of the lens than at the optical center. The radial distortion coefficients model this type of distortion as follows:

$$\begin{aligned} xdistorted &= x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \\ ydistorted &= y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \end{aligned} \tag{5}$$

where:

- x, y is the location of the pixel without distortion.
- k_1 k_2, and k_3 are the radial distortion coefficients of the lens.
- $r^2 = x^2 + y^2$.

On the other hand, tangential distortion occurs when the lenses and the image plane are not perfectly parallel. In this case, the tangential distortion coefficients model this type of distortion as follows:

$$\begin{aligned} xdistorted &= x + \left[p_2\left(r^2 + 2x^2\right) + 2p_1xy\right] \\ ydistorted &= y + \left[p_1\left(r^2 + 2y^2\right) + 2p_2xy\right] \end{aligned} \tag{6}$$

where:

- x, y is the location of the pixel without distortion.
- p_1 and p_2 are the tangential distortion coefficients of the camera.
- $r^2 = x^2 + y^2$.

Using this basis, the computer vision algorithms that have been used in the implementation of the pipeline for the method used to track the plant nutation movement are described in the following points.

2.2.1. Perspective-n-Point (PnP)

The pose computation problem consists of solving the rotation and translation of an object with respect to the camera, while minimizing the reprojection error for correspondences between the 3D points in the world and the 2D points on the image. Several methods have been proposed to estimate the pose of an object relative to the camera [18,19]. These methods can be used for calibration using a planar calibration object [20].

2.2.2. Calibration of a Stereo Pair System

The procedure outlined in the previous subsection enables the obtaining of the intrinsic parameters of the camera as well as the camera pose relative to the calibration object. If the calibration object is visible to both cameras, we have the pose (R_1, t_1) of the first camera relative to the calibration object and the pose (R_2, t_2) of the second camera. These two poses are related as follows:

$$R_2 = R \times R_1 \; ; \; t_2 = R \times t_1 + t \tag{7}$$

Therefore, by using the previous equation, it is possible to obtain the pose of the first camera relative to the second camera. Once the pose of the first camera relative to the second one is obtained, the essential matrix can be constructed as follows:

$$E = \begin{bmatrix} 0 & -t_2 & -t_1 \\ t_2 & 0 & -t_0 \\ -t_1 & t_0 & 0 \end{bmatrix} \times R \tag{8}$$

where t_i are the components of the translation vector $t = [t_0, t_1, t_2]$. From this, the fundamental matrix can be computed as follows:

$$F = CameraMatrix^{-T} \times E \times CameraMatrix^{-1} \tag{9}$$

2.2.3. Background Segmentation

The background segmentation problem involves detecting changes that occur in a sequence of images. Pixels in an image are classified either as part of the background or as part of an object of interest based on a background model. In real applications, depending on the characteristics of the problem, different methods have been proposed: models based on color thresholding, clustering, and fuzzy logic, and, more recently, neural network-based models [21] have been used. The usual steps in image segmentation are:

1. The background model is initialized with some of the video frames.
2. Once the background model is initialized, each pixel in the image is classified based on the background model.
3. The background model is updated with information from the last processed image.

In the algorithms developed to extract the position of the plant's tip in time-lapse images, the Gaussian mixture method was used. This method uses multiple Gaussian functions per pixel to model the background of the scene. This method takes into account that a pixel can have different states over time. The implementation of this method in the OpenCV library is based on Zivkovic [22].

2.3. Motion Tracking and 3D Position Estimation

This section first describes the procedure used to calibrate the stereo system. Next, the steps in the process of extracting the position of the plant tip from the images captured by the stereo system cameras are outlined. Finally, it describes how the nutation motion of the plant was tracked based on the 2D position of the extracted plant tip from the images.

2.3.1. Camera Calibration

The stereo system calibration was carried out following the procedure described in the previous subsection. The calibration process was conducted in two phases. First, each of the stereo cameras was individually calibrated. A calibration object with a 10 × 7 chessboard pattern and dimensions of 59.4 × 84.1 cm was used for the calibration. These dimensions were chosen to cover a large part of the image plane, allowing the precise estimation of the distortion coefficients.

Once the individual calibration of the cameras was completed, the stereo system calibration was performed to estimate the projection matrices for each camera and the fundamental matrix. In this case, a calibration object with dimensions of 42.0 × 59.4 cm was used, visible in both cameras for a sufficient number of poses to carry out the stereo system calibration. Table 2 shows the results of the calibration, measured in terms of the reprojection error, that is, the distance from the calibration pattern key points detected in the images to a corresponding world point projected into the same images.

Table 2. Results of the stereo system calibration. The mean reprojection error (Mean RE) and the standard deviation of the reprojection error (Stdev RE) are shown for the top camera, the side camera, and the complete stereo system.

Reprojection Errors (in Pixels)						
Scenario	**Mean RE Top Camera**	**Stdev RE Top Camera**	**Mean RE Side Camera**	**Stdev RE Side Camera**	**Mean RE Stereo System**	**Stdev RE Stereo System**
No Pole	0.47202	0.34563	0.29831	0.27472	0.61930	0.55591
Pole	0.30597	0.20053	0.28490	0.18405	0.91065	0.80017

In Figure 4, the points used in the calibration of the stereo system can be observed, combining the positions of all the images used for calibration.

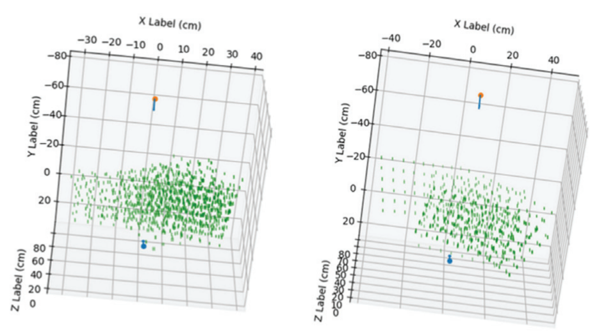


Figure 4. Points used in calibration of the stereo system (in green). **Left**: calibration scenario without pole. **Right**: calibration scenario with pole. The small red and blue arrows indicate the camera's position in the scene.

2.3.2. Detection of the Plant Tip in the Overhead View

The process consisted of two steps, as shown in Figure 5. First, for each frame, *n* points, where the plant tip could be located, were detected. Once the candidate points were obtained, a selection process for the position of the plant tip was performed for each frame, followed by interpolation for frames where no candidate points were detected. The steps of the first step can be seen in Figure 6.

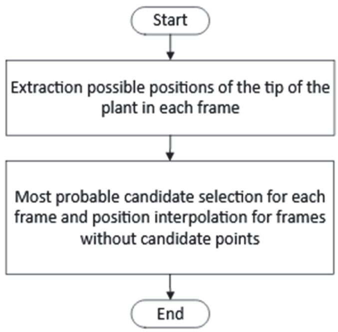


Figure 5. Procedure designed for extracting the plant tip position.

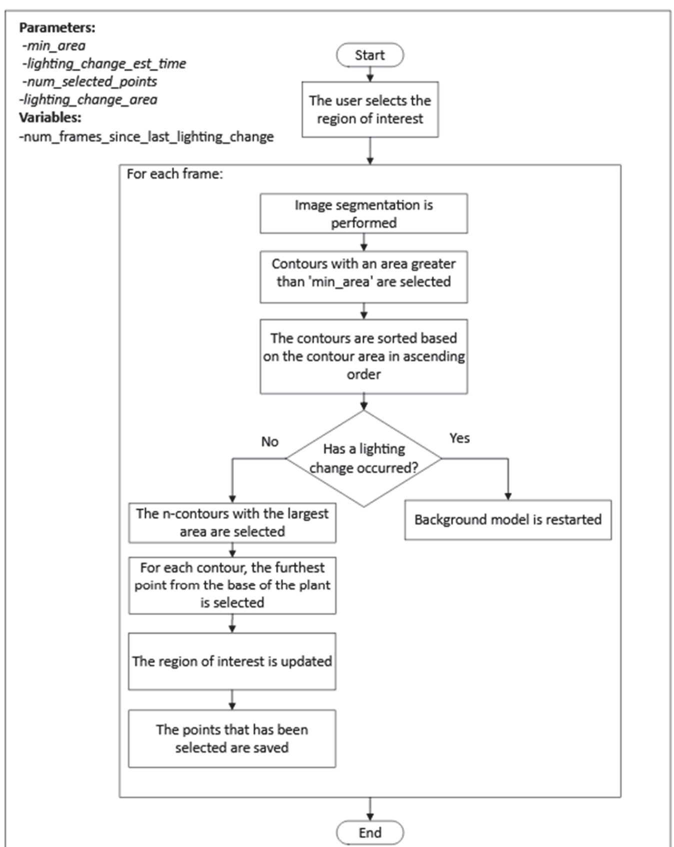


Figure 6. Algorithm for extracting possible positions of the plant tip in each frame of the top view.

The following are the highlights of the algorithm used in the first step of the process:

- The segmentation of the plant contour was performed using the mixture of Gaussians algorithm.
- After each change in lighting, the process reset the background model to reduce the stabilization time.
- To detect when a lighting change occurred, the process checked the area of the selected contours, and, if the area exceeded a threshold specified by the *lighting_change_area* parameter, it was assumed that a lighting change had occurred.
- After a lighting change, the process did not check again for a lighting change until after the frame period specified by the *lighting_change_est_time* parameter to prevent the repeated detection of a lighting change in every frame after the background model had been reset.

The process for selecting the plant tip position among the candidate points followed these two criteria:

1. If there was only one candidate point, it was selected.
2. If there was more than one candidate point, the point closest to the last point selected for previous frames was selected. In segments of frames where there was no candidate point, the position of the plant tip was estimated by performing linear interpolation using the plant tip position in the previous and subsequent frames of the segment.

2.3.3. Detection of the Plant Tip in the Lateral View

This process consisted of two stages, as in the case of the process used for extracting the position of the plant tip in the top view images. First, for each frame, the n points, where the plant tip could be located, were detected. Once the candidate points were obtained, a selection process for the plant tip position was executed for each frame, followed by interpolation for frames where no candidate points were detected. The steps of the first stage are depicted in Figure 7.

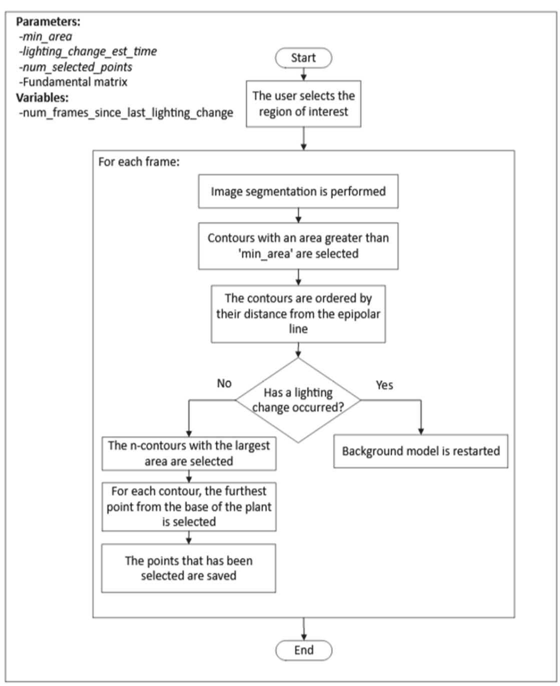


Figure 7. Algorithm for extracting possible positions of the plant tip in each frame of the side view.

The highlights of this algorithm used in the first stage of the process are:

- The segmentation of the plant contour was performed using the mixture of Gaussians algorithm, just as for the segmentation of the plant contour in the top view.
- For detecting changes in lighting, the same procedure used in processing images from the top view was followed.
- There were cases where multiple fragments of the plant contour are obtained when applying the background model to the image. In these cases, the use of the epipolar line helped to identify the fragment where the plant tip was located.
- Once the contours near the epipolar line were selected, a minimum path algorithm was used to select the potential position of the plant tip within the contour.

The process for selecting the plant tip position among the candidate points followed the same criteria as indicated in the previous section.

2.3.4. Correcting Wrong Detections and Obtaining the 3D Positions of the Plant Tip

The algorithms described in the previous subsections were not always able to detect the plant tip position in all cases, as mentioned in each section. For this reason, a "Plant Tracker" tool was developed to allow the user to modify the automatically performed plant tip extractions. This tool consists of three screens, as shown in Figure 8. Once the user has completed the manual corrections, the application allows the extraction of a CSV file with the estimation of the 3D position for a selected range of frames.

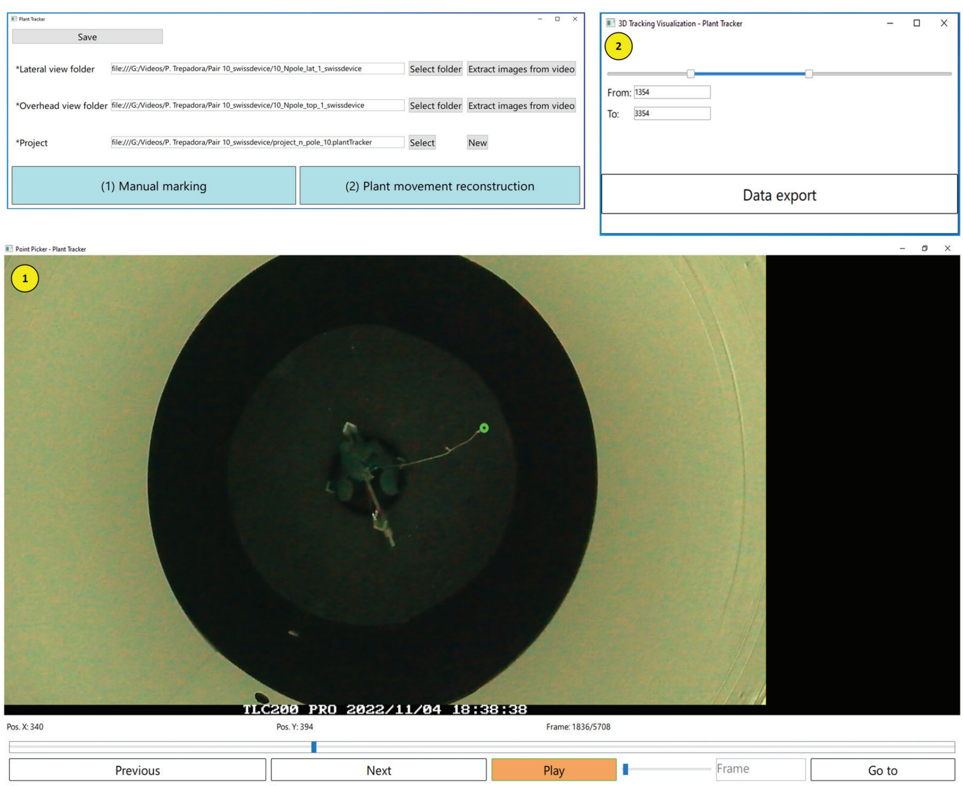


Figure 8. Sample view of the Plant Tracker tool developed to supervise the obtained locations. Top left: main window of the program; button "(1) Manual marking" opens the window labeled 1, and button "(2) Plant movement reconstruction" opens the window labeled 2.

3. Results and Discussion

In this section, the results obtained in the processing of the available videos are presented. The videos were processed in pairs, corresponding to the zenithal and side views of each same experiment. The nutation movement followed by the tip of the bean plant in three of the conducted experiments is shown in Figure 9.

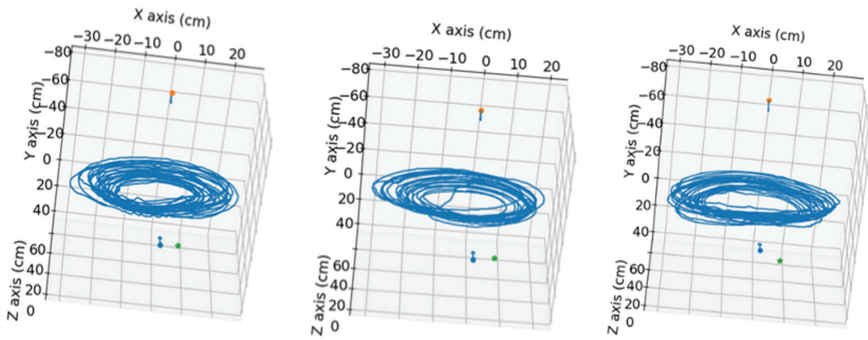


Figure 9. Visualization of the obtained nutation movement of the bean plants. **Left**: position result for video pair 4 without pole. **Middle**: position result for video pair 7 without pole. **Right**: position result for video pair 10 without pole. Red and blue arrows indicate the camera's position in the scene, and the green dot marks the position of the pot's base.

3.1. Results of the Detection of the Tip of the Plant

Table 3 shows the cumulative results of the type of detection used to extract the position of the plant tip in the frames of the time-lapse videos corresponding to the experiments. The types of detection correspond to the following categories: **Automatic:** The algorithm for extracting the plant tip position found only one candidate point. **Manual:** The points that the user had to correct. **Estimated:** The algorithm for extracting the plant tip position found more than one candidate point and selected the one that represented the most probable position of the plant tip. **Interpolated:** The algorithm for extracting the plant tip position did not find a candidate point. The plant tip position in the frame was determined by interpolating the position from the plant position in the preceding and subsequent frames.

Table 3. Results of the plant tip detection procedure in the images for the three video pairs. For each video pair, the condition (with or without a supporting pole), the frames that were analyzed, the camera (top or lateral), and the extraction of the tip location (automatic, manual, estimated or interpolated location) are indicated.

				Type of Detection			
Video	**Pole**	**Frames** [1]	**View**	**Automatic**	**Manual**	**Estimated**	**Interpolated**
Pair 4	No	2483–4746	Lateral	1586	331	293	54
			Top	1878	74	265	47
Pair 7	Yes	3850–5798	Lateral	1607	272	246	100
			Top	2063	27	113	22
Pair 7	No	750–2600	Lateral	1285	208	287	71
			Top	1724	53	74	0
Pair 10	No	3850–5798	Lateral	1353	251	236	109
			Top	1570	151	209	19

[1] Range of frames that have been processed from the video.

Figure 10 shows the reprojection error in pixels for each frame of the videos (side and top views) of the time-lapse within the range for which the 3D position estimation of the plant tip was performed.

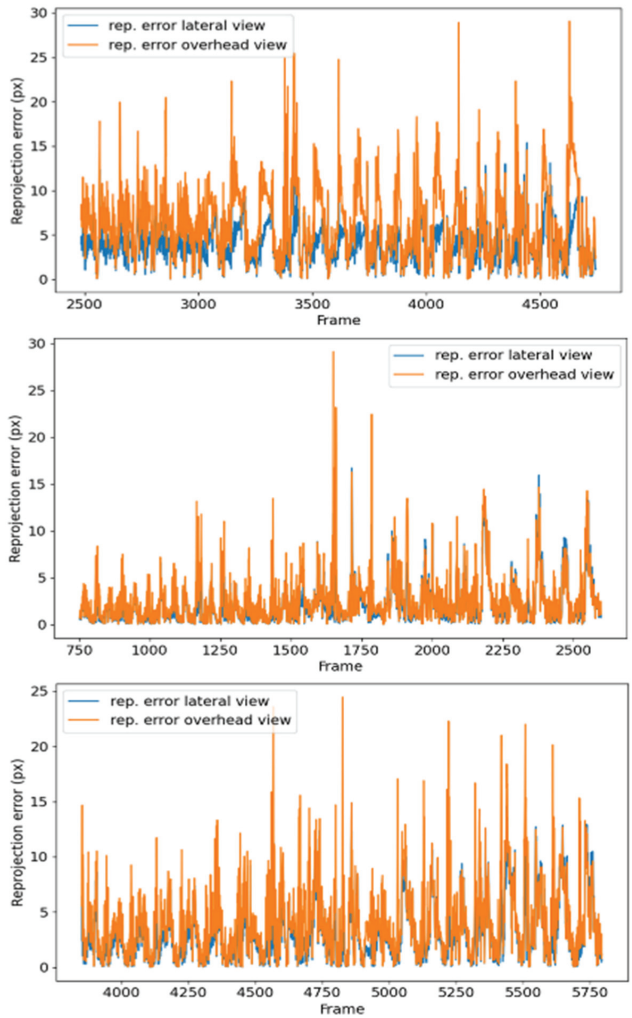


Figure 10. Visualization of reprojection errors of some videos along time. Top row: errors for video pair 4 without pole. Middle: errors for video pair 7 without pole. Bottom row: errors for video pair 10 without pole.

3.2. Discussion of the Results

The plant tip detection algorithms were able to extract the correct position in a range of 86–98% of the cases, as observed in Table 3. The number of corrections made by the user was higher for the plant positions extracted in the lateral plant videos. This was mainly due to moments when the plant tip overlapped with the plant stem in the images. An example of this case is shown in Figure 11. This error is due to the use of a minimum path algorithm to find the plant tip in the segmented contour of the plant. When the plant tip overlapped with the stem, the position farthest from the plant's base corresponded to the elevated part of the plant stem visible in the image.

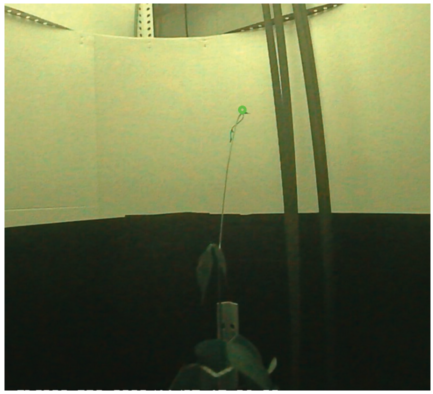


Figure 11. A sample of an incorrect detection of the plant tip in the side view image.

Regarding the reprojection errors, the graphs shown in Figure 10 follow a sawtooth pattern across all processed videos, with the maximum reprojection errors aligned in both the top and side view videos of each time-lapse. Figure 12 shows the positions where the reprojection error is greater than 10 px.

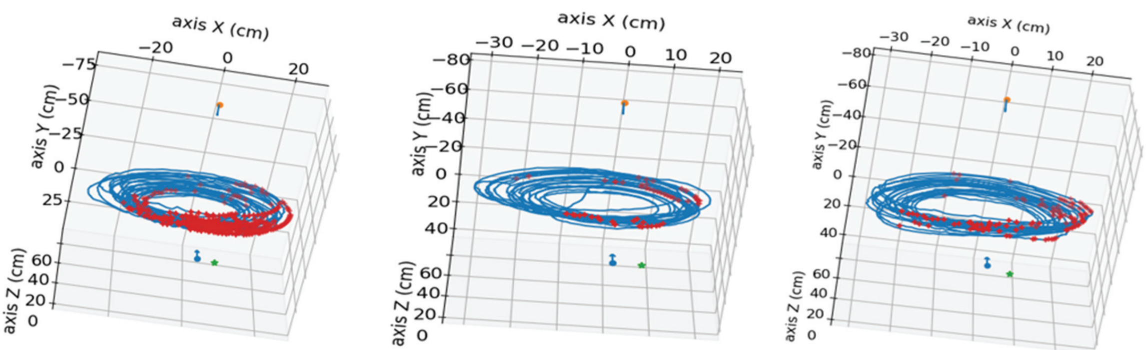


Figure 12. Visualization of the obtained 3D points in the analyzed videos, represented in a blue color. The points with a reprojection error greater than 10 px are marked in red. Red and blue arrows indicate the camera's position in the scene, and the green dots mark the position of the pot's base. **Left**: video pair 4 without pole. **Middle**: video pair 7 without pole. **Right**: video pair 10 without pole.

The average reprojection error for each of the analyzed videos is shown in Table 4.

In general, it can be considered that the proposed method is able to achieve excellent results, obtaining an average reprojection error of only 3.7 px, which is below 0.3% of the width of the images. This value can be roughly translated into an estimated average error in the 3D localization of about 0.5 cm, although this measurement depends on other factors, such as the position of the plant tip. The total number of frames where there is a high error (greater than 10 px) is less than 5.4%. On the other hand, although the method required manual correction by an operator during some frames, as we have seen, this only occurred in 8.3% of the frames. In the remaining 91.7% of the frames, the automatic algorithm worked to locate the tip of the plants. Thus, the accuracy and robustness obtained by the proposed method in tracking the plant tips is suitable for practical use in circumnutation movement experimentation.

Table 4. Results of the reprojection errors for the three video pairs. The mean reprojection error (Mean RE), the standard deviation of the reprojection error (Stdev RE), the total number of points analyzed, and the number of points with a reprojection error greater than 10 px (RE Points > 10 px) are shown for each video pair.

Reprojection Errors (in Pixels)					
Video	**View**	**Mean RE**	**Stdev RE**	**Total Points**	**RE Points > 10 px**
Pair 4	Lateral	3.98679	2.84289	2264	39
	Top	6.49004	5.74389	2264	373
Pair 7	Lateral	2.05453	2.52104	1851	33
	Top	2.66366	3.13255	1851	47
Pair 10	Lateral	2.99391	2.62791	1949	41
	Top	4.08886	3.31717	1949	119

The direct comparison of our results with other methods is not feasible, since other works measure their results in terms of different parameters. For example, in the "Circumnutation Tracker" software of Stolarz et al. [11], manual annotation of the positions is required for all the frames of the videos. Tenzer and Clifford's [12] plant monitoring method only segments the plants, without locating the 3D position of the tip, so the accuracy is given in terms of the area under the ROC curve for the classification problem. In the inertial tracker proposed by Geldhof et al. [15], the errors are expressed as the precision of pitch and roll angles, being under 0.5°. The recent method by Mao et al. [13] using deep learning is able to achieve an average error of 1.02 mm in the location of the apex of the plant, in videos of 640 × 480 px of resolution; however, only one video is analyzed per plant, so the 3D locations are not extracted.

The largest reprojection errors in our method are found in the area closest to the lateral camera. By overlaying the points used in the calibration onto the trace corresponding to the nutation movement of the plant tip, it is observed that not all the space travelled by the plant tip was covered. Furthermore, the greatest reprojection errors are located in the area where there are no calibration points, as shown in Figure 13.

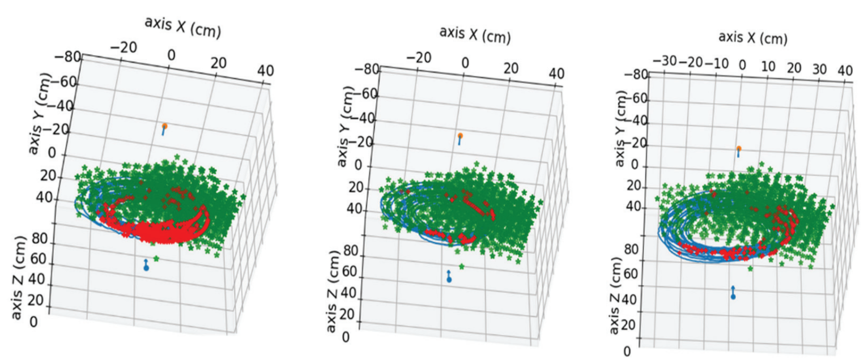


Figure 13. Visualization of 3D points with a reprojection error greater than 10 px (marked in red) and calibrations points (marked in green). **Left**: video pair 4 without pole. **Middle**: video pair 7 without pole. **Right**: video pair 10 without pole. Red and blue arrows indicate the camera's position in the scene, and the green dots mark the position of the pot's base.

To prevent this error in future experiments, it will be verified during the calibration process that there are enough calibration points across the entire space traveled by the plant tip in its nutation movement.

4. Conclusions

The study of the nutation movement of plants has aroused great interest in the research community in different areas, from philosophy to computer science, and has been a great source of inspiration for new algorithms and bioinspired systems. This paper has presented the development of a stereo vision system for studying the movement of climbing plants. The system consists of a pair of RGB cameras that synchronously record a side and top view of the plants in time-lapse. This study is focused on the common bean as a typical climbing plant model. Subsequently, the images are analyzed with a computer vision algorithm that obtains the tip of the plant and, using a previous calibration of the stereo pair, estimates the position in 3D coordinates. The method is able to extract the correct tip position in 86–98% of cases, depending on the video, with an average reprojection error below 4 px, which is translated to an approximate error in the 3D localization of about 0.5 cm. The proposed method allows researchers to know precisely and robustly the nutation movements of the plants and to compare their behavior under different situations, such as the use or absence of support structures for climbing.

In future work, it would be interesting to apply the latest deep learning methods to perform the accurate segmentation of the plants in the images, as well as subsequent matching between the stereo pair images for a complete estimation of the 3D position of the plant. In this way, it would be possible to know not only the position of the tip of the plant, but also of other parts such as the stems and leaves.

Author Contributions: Conceptualization, P.C.; methodology, A.P., P.C. and G.G.-M.; software, D.R.R.-M.; validation, D.R.R.-M. and G.G.-M.; investigation, D.R.R.-M., A.P., P.C. and G.G.-M.; resources, P.C. and A.P.; data curation, D.R.R.-M.; writing—original draft preparation, D.R.R.-M. and G.G.-M.; writing—review and editing, P.C.; visualization, D.R.R.-M.; supervision, P.C. and G.G.-M.; project administration, P.C.; and funding acquisition, P.C. and G.G.-M. All authors have read and agreed to the published version of this manuscript.

Funding: This research was partly funded by project 22130/PI/22 financed by the Region of Murcia (Spain) through the Regional Program for the Promotion of Scientific and Technical Research of Excellence (Action Plan 2022) of the Fundación Séneca-Agencia de Ciencia y Tecnología de la Región de Murcia and partly by the Office of Naval Research Global (Award # N62909-19-1-2015), awarded to P.C. and A.P.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data used in this study are available from the authors upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of this study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Mugnai, S.; Azzarello, E.; Masi, E.; Pandolfi, C.; Mancuso, S. Nutation in Plants. In *Rhythms in Plants: Dynamic Responses in a Dynamic Environment*; Springer: Cham, Switzerland, 2015; pp. 19–34. [CrossRef]
2. Raja, V.; Silva, P.L.; Holghoomi, R.; Calvo, P. The dynamics of plant nutation. *Sci. Rep.* **2020**, *10*, 19465. [CrossRef] [PubMed]
3. Gilroy, S. Plant tropisms. *Curr. Biol.* **2008**, *18*, R275–R277. [CrossRef] [PubMed]
4. Sperotto, P.; Acevedo-Rodríguez, P.; Vasconcelos, T.N.; Roque, N. Towards a standardization of terminology of the climbing habit in plants. *Bot. Rev.* **2020**, *86*, 180–210. [CrossRef]
5. Darwin, C.; Darwin, F. *The Power of Movement in Plants*; John Murray: London, UK, 1880.
6. Gallentine, J.; Wooten, M.B.; Thielen, M.; Walker, I.D.; Speck, T.; Niklas, K. Searching and intertwining: Climbing plants and GrowBots. *Front. Robot. AI* **2020**, *7*, 118. [CrossRef]
7. Meroz, Y. Plant tropisms as a window on plant computational processes. *New Phytol.* **2021**, *229*, 1911–1916. [CrossRef]
8. Calvo, P.; Lawrence, N. *Planta Sapiens: Unmasking Plant Intelligence*; W. W. Norton & Company: Hachette, UK, 2022.
9. Burris, J.N.; Lenaghan, S.C.; Stewart, C.N. Climbing plants: Attachment adaptations and bioinspired innovations. *Plant Cell Rep.* **2018**, *37*, 565–574. [CrossRef]

10. Navarro, P.J.; Fernández, C.; Weiss, J.; Egea-Cortines, M. Development of a configurable growth chamber with a computer vision system to study circadian rhythm in plants. *Sensors* **2012**, *12*, 15356–15375. [CrossRef] [PubMed]
11. Stolarz, M.; Żuk, M.; Król, E.; Dziubińska, H. Circumnutation Tracker: Novel software for investigation of circumnutation. *Plant Methods* **2014**, *10*, 24. [CrossRef]
12. Tenzer, M.L.; Clifford, N.C. A digital green thumb: Neural networks to monitor hydroponic plant growth. In Proceedings of the 2020 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 24 April 2020; pp. 1–6.
13. Mao, Y.; Liu, H.; Wang, Y.; Brenner, E.D. A deep learning approach to track Arabidopsis seedlings' circumnutation from time-lapse videos. *Plant Methods* **2023**, *19*, 18. [CrossRef] [PubMed]
14. Díaz-Galián, M.V.; Perez-Sanz, F.; Sanchez-Pagán, J.D.; Weiss, J.; Egea-Cortines, M.; Navarro, P.J. A proposed methodology to analyze plant growth and movement from phenomics data. *Remote Sens.* **2019**, *11*, 2839. [CrossRef]
15. Geldhof, B.; Pattyn, J.; Eyland, D.; Carpentier, S.; Van de Poel, B. A digital sensor to measure real-time leaf movements and detect abiotic stress in plants. *Plant Physiol.* **2021**, *187*, 1131–1148. [CrossRef] [PubMed]
16. Guerra, S.; Peressotti, A.; Peressotti, F.; Bulgheroni, M.; Baccinelli, W.; D'Amico, E.; Gómez, A.; Massaccesi, S.; Ceccarini, F.; Castiello, U. Flexible control of movement in plants. *Sci. Rep.* **2019**, *9*, 16570. [CrossRef] [PubMed]
17. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press eBooks: Cambridge, UK, 2003. Available online: https://digitalcollections.anu.edu.au/handle/1885/52332 (accessed on 21 January 2024).
18. Zhang, Z.P. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [CrossRef]
19. Lepetit, V.; Moreno-Noguer, F.; Fua, P. EPnP: An Accurate O(n) Solution to the PnP Problem. *Int. J. Comput. Vis.* **2008**, *81*, 155–166. [CrossRef]
20. Hesch, J.A.; Roumeliotis, S.I. A Direct Least-Squares (DLS) method for PnP. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 383–390. [CrossRef]
21. Garcia-Garcia, B.; Bouwmans, T.; Rosales, A. Background subtraction in real applications: Challenges, current models and future directions. *Comput. Sci. Rev.* **2020**, *35*, 100204. [CrossRef]
22. Zivkovic, Z. Improved adaptive Gaussian mixture model for background subtraction. In Proceedings of the Pattern Recognition, ICPR 2004, of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26 August 2004. [CrossRef]

sensors

MDPI

Article

Feasibility of 3D Body Tracking from Monocular 2D Video Feeds in Musculoskeletal Telerehabilitation

Carolina Clemente [1,2,*], Gonçalo Chambel [2], Diogo C. F. Silva [3,4,5], António Mesquita Montes [3,5,6], Joana F. Pinto [2] and Hugo Plácido da Silva [1,7,8,*]

1 Instituto Superior Técnico (IST), Department of Bioengineering (DBE), Av. Rovisco Pais n. 1, 1049-001 Lisboa, Portugal
2 CLYNXIO , LDA, Rua Augusto Macedo, n. 6, 5 Dto., 1600-794 Lisboa, Portugal
3 Department of Physiotherapy, Santa Maria Health School, Trav. Antero de Quental 173/175, 4049-024 Porto, Portugal; diogo.silva@santamariasaude.pt (D.C.F.S.); antonio.montes@santamariasaude.pt (A.M.M.)
4 Department of Functional Sciences, School of Health, Polytechnic Institute of Porto, Rua Dr. António Bernardino de Almeida 400, 4200-072 Porto, Portugal
5 Center for Rehabilitation Research, School of Health, Polytechnic Institute of Porto, Rua Dr. António Bernardino de Almeida 400, 4200-072 Porto, Portugal
6 Department of Physiotherapy, School of Health, Polytechnic Institute of Porto, Rua Dr. António Bernardino de Almeida 400, 4200-072 Porto, Portugal
7 Instituto de Telecomunicações (IT), Av. Rovisco Pais n. 1, Torre Norte—Piso 10, 1049-001 Lisboa, Portugal
8 Lisbon Unit for Learning and Intelligent Systems (LUMLIS), European Laboratory for Learning and Intelligent Systems (ELLIS), Av. Rovisco Pais n. 1, 1049-001 Lisboa, Portugal
* Correspondence: carolina.v.clemente@tecnico.ulisboa.pt (C.C.); hsilva@lx.it.pt (H.P.d.S.)

Abstract: Musculoskeletal conditions affect millions of people globally; however, conventional treatments pose challenges concerning price, accessibility, and convenience. Many telerehabilitation solutions offer an engaging alternative but rely on complex hardware for body tracking. This work explores the feasibility of a model for 3D Human Pose Estimation (HPE) from monocular 2D videos (MediaPipe Pose) in a physiotherapy context, by comparing its performance to ground truth measurements. MediaPipe Pose was investigated in eight exercises typically performed in musculoskeletal physiotherapy sessions, where the Range of Motion (ROM) of the human joints was the evaluated parameter. This model showed the best performance for shoulder abduction, shoulder press, elbow flexion, and squat exercises. Results have shown a MAPE ranging between 14.9% and 25.0%, Pearson's coefficient ranging between 0.963 and 0.996, and cosine similarity ranging between 0.987 and 0.999. Some exercises (e.g., seated knee extension and shoulder flexion) posed challenges due to unusual poses, occlusions, and depth ambiguities, possibly related to a lack of training data. This study demonstrates the potential of HPE from monocular 2D videos, as a markerless, affordable, and accessible solution for musculoskeletal telerehabilitation approaches. Future work should focus on exploring variations of the 3D HPE models trained on physiotherapy-related datasets, such as the Fit3D dataset, and post-preprocessing techniques to enhance the model's performance.

Keywords: telerehabilitation; musculoskeletal; 3D Human Pose Estimation; MediaPipe Pose; ROM; 2D camera; monocular; videos; deep learning

Citation: Clemente, C.; Chambel, G.; Silva, D.C.F.; Montes, A.M.; Pinto, J.F.; Silva, H.P.d. Feasibility of 3D Body Tracking from Monocular 2D Video Feeds in Musculoskeletal Telerehabilitation. *Sensors* **2024**, *24*, 206. https://doi.org/10.3390/s24010206

Academic Editors: Fernando Fernández-Martínez, Manuel Gil-Martín and Rubén San-Segundo

Received: 22 November 2023
Revised: 20 December 2023
Accepted: 21 December 2023
Published: 29 December 2023

1. Introduction

Musculoskeletal conditions are the leading cause of physiotherapy demand globally; it is estimated that approximately 1.71 billion people have a musculoskeletal condition worldwide, and its prevalence is expected to increase [1]. Conventional musculoskeletal rehabilitation typically involves multiple in-clinic sessions, high travelling and waiting times, little schedule flexibility and complementary home exercises, becoming inconvenient and expensive for both patients and clinics [2]. Consequently, providing quality care to a patient at a distance (i.e., telerehabilitation) through digital and gamified solutions has become a topic of growing interest [2].

The development of telerehabilitation has increased in recent years, due to factors including technological innovation, the impact of the COVID-19 pandemic, and the rising concern about patient-centered treatment [3,4]. Telerehabilitation has been demonstrated to offer enhanced accessibility, affordability, personalization, and real-time monitoring; furthermore, it empowers patients to take charge of their well-being while ensuring convenience, adherence, and motivation [3,5]. Moreover, by reducing wait times, facilitating long-term management, and fostering patient education, telerehabilitation is emerging as an indispensable component of modern healthcare, reshaping the future of physiotherapy services [4]. Musculoskeletal telerehabilitation should resemble traditional sessions and allow for equivalent outcomes compared to conventional methods, in order to ensure its widespread acceptance and integration into healthcare systems [3]. Human body tracking is, therefore, a key element, since it allows physiotherapists to be aware of the movement and performance of their patients during the physiotherapy session [6], while following the relevant clinical and motion information, such as the Range of Motion (ROM) of the human joints. The joint ROM is defined as "the amount of movement that occurs at a joint to produce movement of a bone in space", i.e., the angle range of a joint during an exercise [7]. It is a valuable musculoskeletal metric since it provides information to identify limitations and imbalances in joint movement, gives useful knowledge for developing an appropriate treatment, and helps to ensure that exercises are being performed correctly and safely, maximizing the benefits of the exercise and minimizing the risk of injury [7].

Human body tracking systems for musculoskeletal rehabilitation purposes include optical marker-based systems [8], wearables [9,10], markerless 3D or RGB-D (Red Green Blue-Depth) camera-based systems [5,11], and markerless 2D or RGB (Red Green Blue) camera-based systems [12].

Optical marker-based and wearable sensor systems provide highly accurate body tracking but can be expensive and cumbersome [13,14]. On the other hand, 3D and 2D camera-based systems are more convenient for patients as no body-worn markers/sensors are required (markerless); however, 3D cameras may be costly when compared to 2D cameras, often have software restrictions (e.g., Orbbec Astra 3D cameras are incompatible with MacOS X), and depend on calibration and luminosity conditions [15]. Two-dimensional camera-based systems are a promising, accessible, and affordable body tracking approach [16]. These only require images or videos from a regular 2D camera (available in most standard computers and smartphones) to reconstruct the human body, typically through deep learning models [12]. Considering the advantages of 2D camera-based systems, this work aims to evaluate if these approaches are feasible for musculoskeletal telerehabilitation purposes. To the best of our knowledge, this is the first work investigating a 2D camera-based approach for human body tracking in a diverse set of exercises typically used in musculoskeletal telerehabilitation (and conventional) sessions. Furthermore, this work provides an in-depth description of novel methodologies for coordinate system definition, Range of Motion (ROM) calculation, and data alignment between acquisition systems with different frame rates.

The remainder of the work is structured into the following segments. Section 2 describes the background and main concepts of 2D camera-based systems, drawing conclusions about the most suitable model for the purpose (MediaPipe Pose). Section 3 details the experimental study conducted to evaluate the MediaPipe Pose model for 3D body tracking from monocular video feeds. Section 4 summarizes the main experimental results. Section 5 provides the discussion of the study outcomes. Section 6 draws the main findings and conclusions about this work.

2. Background

In the context of 2D camera-based systems using deep learning algorithms, human body tracking is commonly referred to as Human Pose Estimation (HPE), and the human body is typically represented by a skeleton model—a tree-like structure composed of landmarks (human joints and other keypoints) connected by edges (body segments) [12], as shown in Figure 1. The joints include shoulders, elbows, wrists, hips, knees, and ankles. Keypoints from the vertebral column, hands, feet, and face may also be integrated into the pose estimation. This body representation uses relatively low-dimensional parameters and is highly valuable for motion capture techniques [12].

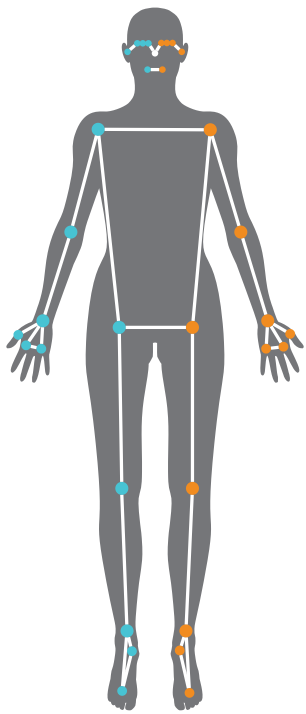


Figure 1. Example of skeletal human body representation: 33 landmarks of MediaPipe Pose , where the right-side landmarks are represented in blue, the left-side landmarks in orange, and the nose landmark in white.

HPE can be classified based on various criteria (Figure 2), and models can be categorized into 2D HPE or 3D HPE. The former calculates the 2D coordinates (x, y) of human keypoints, while the latter determines the 3D coordinates (x, y, z) of human joints. The primary difference lies in the depth coordinate (z). Additionally, 3D HPE can be classified as single-view (or monocular), when a single 2D camera is used in a fixed position, or as multi-view, when two or more 2D cameras are used to observe the person from different viewpoints [12]. When considering the number of people detected, algorithms can be categorized as single-person or multi-person. Single-person approaches involve identifying the keypoints of a single body in the image or video, while multi-person approaches require the model to identify multiple bodies and their respective landmarks, becoming a challenging task. Multi-person approaches are further divided into top-down and bottom-up methods. Top-down methods first detect individual subjects and then estimate the keypoints (and poses), while bottom-up methods first localize body parts across the image and then associate them to assemble complete human poses. Bottom-up approaches tend to have faster inference, however, top-down methods yield higher accuracy [12].

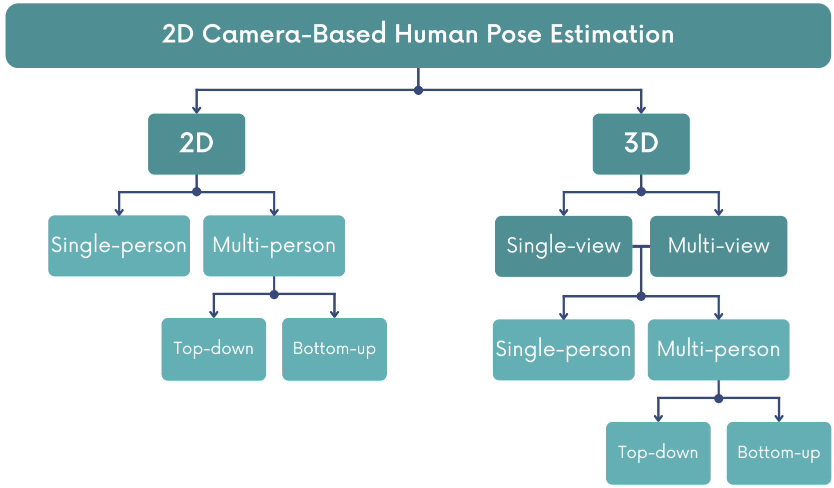


Figure 2. Classification of 2D camera-based models for Human Pose Estimation (HPE).

This work is focused on single-view (or monocular) and single-person approaches. Firstly, monocular approaches require a single camera (fewer resources); furthermore, in telerehabilitation, the treatment sessions are commonly patient-centered, meaning that the camera needs to detect a single subject (single-person approach).

The emergence of deep learning techniques has significantly enhanced the performance of the models for 2D HPE; 3D HPE approaches arise as an extension of 2D HPE [12]. Two of the main challenges of HPE methods are occlusions and rare poses, due to limited training data [12,17]. The sources of occlusions are the person (self-occlusions), other people (in multi-person detections), or external objects [18]. In 2D HPE, numerous approaches have been explored to address occlusions, by adapting the architecture of the deep learning algorithm [12]. In 3D HPE, occlusions can be addressed by collecting information from different viewpoints (a multi-view approach), as an occluded segment in one view may become visible from alternative perspectives [12]. For this work, multi-view approaches are not explored, as the primary goal is to use a single 2D camera for human body tracking.

The performance of 2D HPE has been researched in the literature [12] and various open-source models have been developed, such as PoseNet [19], MoveNet [20], AlphaPose [21], OpenPose [22], and MediaPipe Pose or BlazePose [23]. The evaluation of these models for rehabilitation-related purposes shows encouraging results [24–27]. Three-dimensional HPE models in the literature include VNect [28], XNect [18], and MediaPipe Pose [23,29]. Although other algorithms have been evaluated in previous research [12], their implementations are typically obfuscated and are rarely evaluated in a physiotherapy context [16].

Considering the aim of this project, MediaPipe Pose was selected; it is a deep learning-based model for 3D HPE from monocular 2D videos with promising performance that combines fast performance, reasonable accuracy, and accessibility [23,29]. MediaPipe Pose predicts the 3D central position of the human joints (Figure 1), in meters, which may be used for several purposes, including calculating the Range of Motion (ROM) of the human joints. This metric is extremely valuable in musculoskeletal physiotherapy since physiotherapists use it to assess the mobility and functioning of joints, and monitor the patient's evolution [7]. Furthermore, this parameter evaluates pose accuracy independently from scale and body proportions [30].

3. Materials and Methods

3.1. Experimental Design

An experimental study was designed to evaluate the performance of MediaPipe Pose on a wide range of exercises typically performed in musculoskeletal physiotherapy sessions, by comparing it to a gold standard motion tracking system, namely the Qualisys Motion Capture system (Gothenburg, Sweden; https://www.qualisys.com/, accessed on 3 August 2023). Eight exercises were selected: Shoulder Flexion/Extension (SF), Shoulder Abduction/Adduction (SA), Elbow Flexion/Extension (EF), Shoulder Press (SP), Hip Abduction/Adduction (HA), Squat (SQ), March (MCH), and Seated Knee Flexion/Extension (SKF). The exercises are described in Table 1 and shown in Figure 3. The exercise selection was based on their engagement of diverse joints (shoulder, elbow, hip, and knee), body poses, moving limbs (upper and lower limbs), and planes of movement (frontal and sagittal). A broad range of movements allows the evaluation of the model given different conditions, each presenting various challenges in diversity, complexity, and occlusions.

Our study involved eight healthy participants (seven females and one male) aged between 19 and 21 years old. In order to assess the physical condition and musculoskeletal health of the participants, a questionnaire was designed to gather relevant information, including neuro-musculoskeletal injuries clinically diagnosed in the last three months, movements that elicited pain, and any previous musculoskeletal surgeries. All subjects were deemed eligible and were enrolled in the study. The experimental protocol was approved by the Ethics Committee of the Escola Superior de Saúde de Santa Maria (Reference: CE2022/09). A written informed consent was obtained from all participants.

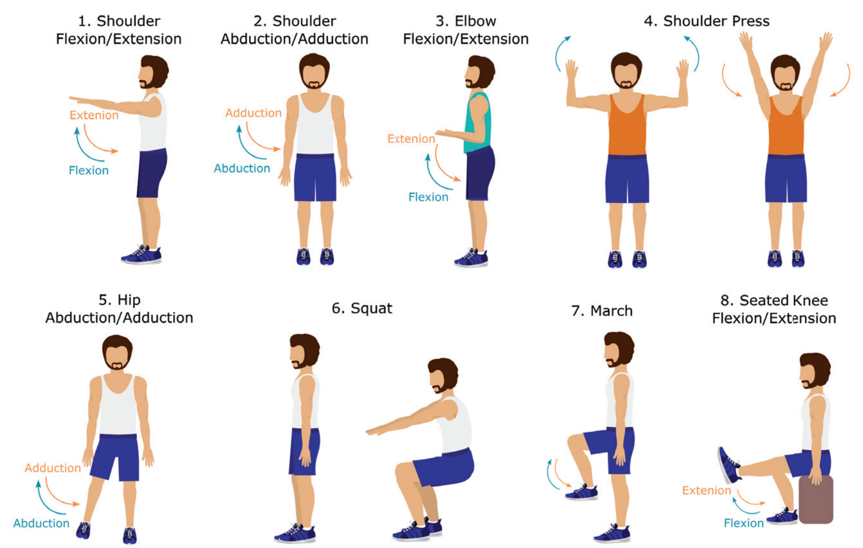


Figure 3. Eight exercises selected for the experimental study: Shoulder Flexion/Extension (SF), Shoulder Abduction/Adduction (SA), Elbow Flexion/Extension (EF), Shoulder Press (SP), Hip Abduction/Adduction (HA), Squat (SQ), March (MCH), and Seated Knee Flexion/Extension (SKF). Shoulder press and squat exercises are illustrated by a sequence of two representative images of the movement.

Table 1. Exercises commonly performed in musculoskeletal physiotherapy sessions, and description of the limb in motion, the plane of movement, and the evaluated joint.

Exercises	Limb in Motion	Plane of Movement	Evaluated Joint
1. Shoulder Flexion/Extension (SF)	Right arm	Sagittal	Right shoulder
2. Shoulder Abduction/Adduction (SA)	Right arm	Frontal	Right shoulder
3. Elbow Flexion/Extension (EF)	Arms (bilateral)	Sagittal	Right elbow
4. Shoulder Press (SP)	Arms (bilateral)	Frontal	Right shoulder
5. Hip Abduction/Adduction (HA)	Right leg	Frontal	Right hip
6. Squat (SQ)	Legs (bilateral)	Sagittal	Right knee
7. March (MCH)	Legs (bilateral)	Sagittal	Right hip
8. Seated Knee Flexion/Extension (SKF)	Right leg	Sagittal	Right knee

3.2. Experimental Data Acquisition

Experimental data acquisition was conducted in the biomechanics laboratory of the Centro de Investigação em Reabilitação (CIR) at the Escola Superior de Saúde do Instituto Politécnico do Porto. Healthy participants performed eight exercises displayed on a screen to guide them through the execution of the movements. Before initiating each exercise, a preview of the movement was shown to illustrate the exercise that the participants were expected to perform. Then, the acquisition consisted of each participant performing two sets of seven exercise repetitions, with a 10-second resting period between sets. Between acquisitions, participants were advised to rest by sitting on a chair before starting the following exercise. A total of 64 acquisitions (eight participants $\times$ eight exercises) were collected, of which only 63 were studied due to a technical issue during the shoulder flexion exercise performed by Subject 5. For each acquisition, the following data were collected simultaneously: (1) ground truth data, i.e., 3D coordinates (x, y, z) of various anatomical landmarks, using the Qualisys Motion Capture system (recording data at 100 frames per second, FPS) from the laboratory; and (2) monocular 2D video recordings, using a Nikon Coolpix A10 camera (operating with 1280 $\times$ 720 resolution at 30 FPS). Figure 4 depicts the experimental setup.

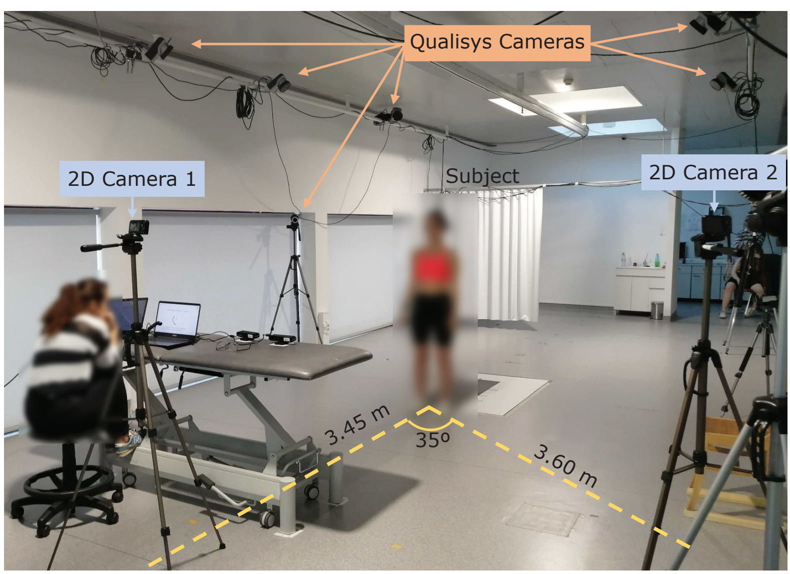


Figure 4. Experimental setup for the data acquisition, showing some of the Qualisys cameras, two 2D cameras, and the relative position between the subject and the two 2D cameras.

Before starting the data acquisition, the Qualisys configuration involved the calibration of the 12 infrared cameras, followed by the placement of motion capture (MoCap) markers on specific anatomical regions of the participants. For this study, data from only six MoCap markers placed on the right side of the body were required. It was essential to determine and establish the correct position of the anthropometric points, as these were used as ground truth measurements; therefore, the anatomical points of the MoCap markers were defined and confirmed by a physiotherapist researcher experienced in palpatory anatomy. Figure 5 shows the anatomical location of the markers, and Table 2 describes the association between the anatomical location of the six Qualisys MoCap markers and the human joints. For the 2D video recording, two identical 2D cameras were used (only one was operating at any one time); 2D camera 1 recorded frontal plane exercises and was parallel to the participant's frontal plane, and 2D camera 2 recorded sagittal plane exercises and was positioned at an angle of 35° to the participant's frontal plane (Figure 4). The previous information describes the camera position that minimizes the number of occlusions during the exercise execution.

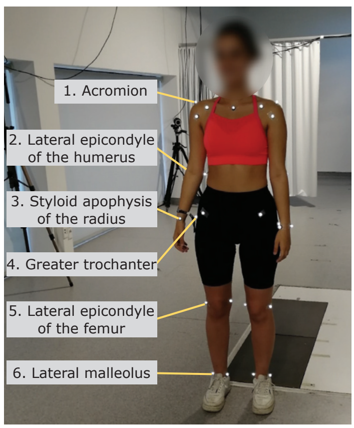


Figure 5. Anatomical location of the six Qualisys MoCap markers.

Table 2. Relation between the anatomical location of the six Qualisys MoCap markers and the human joints.

MoCap Anatomical Location	Joint
1. Acromion	Shoulder
2. Lateral epicondyle of humerus	Elbow
3. Styloid apophysis of radius	Wrist
4. Greater trochanter	Hip
5. Lateral epicondyle of the femur	Knee
6. Lateral malleolus of the ankle	Ankle

3.3. Data Preprocessing

The 2D videos from the experimental acquisition were given as input for the MediaPipe Pose model to estimate the 3D coordinates of the human joints. The raw data from the Qualisys system and the MediaPipe Pose model consisted of the 3D positions of the MoCap markers and the 3D central positions of the joints, respectively. For the ROM evaluation, both data sources provide approximately equivalent information after converting the 3D positions into amplitude values. Additionally, proper alignment was necessary for comparing ground truth and predicted values from the same time point. The procedures are described next.

3.3.1. 3D Cartesian Coordinate System

For physiotherapy purposes, defining a 3D coordinate system coincident with the normal vectors of the anatomical planes is valuable information, particularly when determining the joint amplitude or Range of Motion (ROM).

The Qualisys coordinate system is shown in Figure 6. The direction of the axes of the Qualisys coordinate system was assumed to be parallel to the normal vectors of the anatomical planes of the participant (Figure 7).

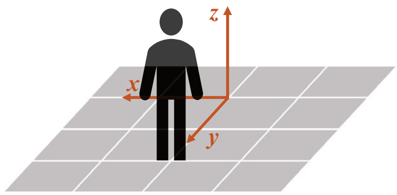


Figure 6. The 3D Cartesian coordinate system of Qualisys (in orange) and its spatial relation with respect to the participant position during data acquisition.

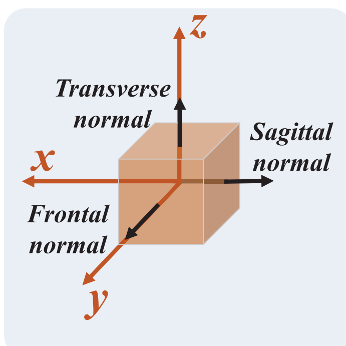
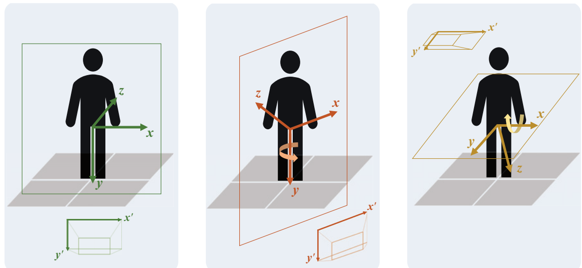


Figure 7. Comparison of the normal vectors of the anatomical planes (in black) with the Qualisys coordinate system (in orange).

The MediaPipe Pose coordinate system depends on the relative position between the 2D camera and the participant, as illustrated in Figure 8. The origin is the midpoint between the hips; the XY plane of the MediaPipe Pose coordinate system is parallel to the X'Y' plane of the camera plane. The Z-axis is the third direction according to the right-hand rule. Due to the camera position dependency, no direct relationship between the MediaPipe Pose and anatomical coordinate systems can be assumed. Therefore, a virtual coordinate system was defined coincident with the anatomical planes, such that the Z-axis is the normal vector to the frontal plane, the X-axis the normal vector to the sagittal plane, and the Y-axis the normal vector to the transverse plane (Figure 9).

Figure 10 shows the three main steps to define the virtual coordinate system. The origin is the midpoint between the hips, and the direction of the axes was defined using the torso joint positions (shoulders and hips) estimated by MediaPipe Pose in the first frame. Firstly, the frontal plane normal (Z-axis) was defined as the normal vector to the best plane containing the four torso keypoints, using the RANSAC regressor [31]. Secondly, the transverse plane normal (Y-axis) was defined as the vector from the shoulders' midpoint to the hips' midpoint. Lastly, the sagittal plane normal (X-axis) was calculated using the right-hand rule. The direction of the axes of the MediaPipe Pose virtual coordinate system was assumed to be parallel to the normal vectors of the anatomical planes of the participant (Figure 11).



(**a**) (**b**) (**c**)

Figure 8. Relation between the participant position and the Cartesian coordinate system of the MediaPipe Pose model for three camera orientations: (**a**) camera plane parallel to participant frontal plane; (**b**) camera plane rotated around the Y-axis relative to participant frontal plane; and (**c**) camera plane rotated around the X-axis relative to participant frontal plane. The camera 2D coordinate system is represented by the X′-axis and Y′-axis, which are parallel to the X-axis and Y-axis of the algorithm coordinate system, respectively.

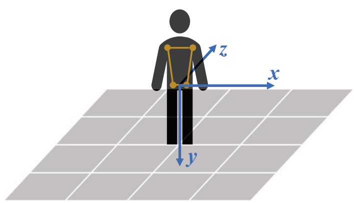


Figure 9. The virtual 3D coordinate system of MediaPipe Pose coincident with the normal vectors of the anatomical planes. The origin is the midpoint between the hips. The X-axis is the sagittal plane normal, the Y-axis is the transverse plane normal, and the Z-axis is the frontal plane normal. The four points (representing the shoulders and hips) are used to define the virtual 3D coordinate system.

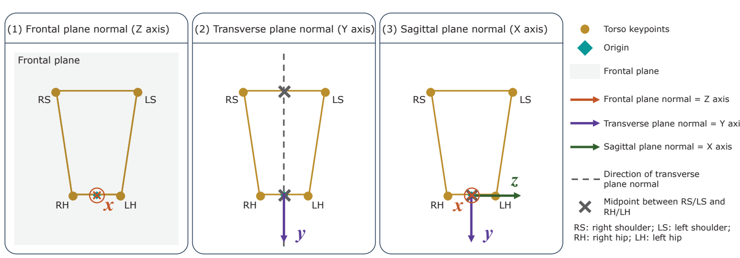


Figure 10. Representation of virtual 3D coordinate system definition: (1) Z-axis or frontal plane normal; (2) Y-axis or transverse plane normal; and (3) X-axis or sagittal plane normal.

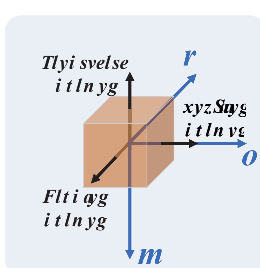


Figure 11. Comparison of the normal vectors of the anatomical planes (in black) with the MediaPipe Pose virtual coordinate system (in blue).

3.3.2. Amplitude Calculation

In musculoskeletal physiotherapy, the Range of Motion (ROM) is defined according to the neutral-zero method; the ROM is determined by moving the distal segment of a joint from a neutral starting position (zero position) to the end position around a rotation axis [7]. Therefore, the joint amplitude is the angle between the body segment (projected in the plane of movement) and a reference direction (zero position), as shown in Figure 12. The body segment is the vector between two joints: the joint where the ROM is being evaluated and the closest joint to the evaluated joint in the moving limb. The projection of the body segment of the plane of movement ensures that only the angle component of that plane is being investigated. For instance, for the shoulder abduction/adduction exercise, only the frontal plane component assesses the shoulder joint on abduction/adduction motions; the sagittal plane component evaluates it on flexion/extension motions. The normal to the plane of movement is the only information necessary for the projection of the body segment on that plane. The reference direction is the vector with respect to which the amplitude is defined, meaning that it corresponds to the 0° amplitude (zero position). The information for the amplitude calculation for each exercise is shown in Table 3.

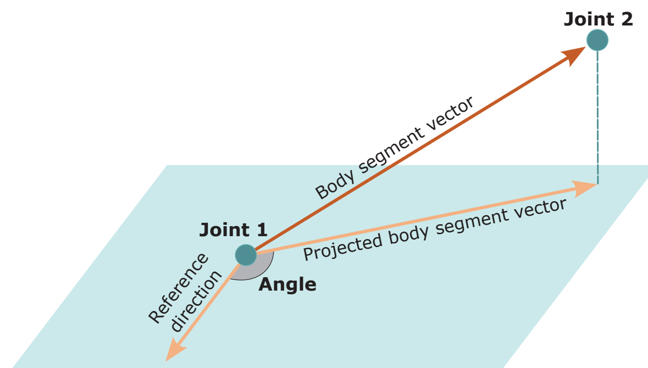


Figure 12. Amplitude calculation between the projected body segment vector and a reference direction.

Table 3. Information for the ROM evaluation: plane of movement in which the exercise occurs, the body segment, and the reference direction. ↓ represents vertically downward direction.

Exercises	Plane of Movement	Body Segment (Joint 1–Joint 2)	Reference Direction
1. SF	Sagittal	Shoulder–elbow	↓
2. SA	Frontal	Shoulder–elbow	↓
3. EF	Sagittal	Elbow–wrist	↓
4. SP	Frontal	Shoulder–elbow	↓
5. HA	Frontal	Hip–knee	↓
6. SQ	Sagittal	Knee–hip	Foot-knee
7. MCH	Sagittal	Hip–knee	↓
8. SKF	Sagittal	Knee–foot	↓

3.3.3. Data Alignment

After determining the raw amplitude data, finding matching time points between Qualisys and MediaPipe Pose amplitudes was required. This was achieved by implementing five steps (Figure 13): (1) converting frames into a time scale, in seconds, knowing the frame rate of Qualisys (100 FPS) and 2D videos (30 FPS); (2) finding maximum (peak) amplitudes, and the respective time points; (3) aligning amplitude acquisitions by the first peak time point; (4) downsampling Qualisys time points by selecting the ones that matched MediaPipe Pose time points; and (5) fine-tuning the alignment by selecting the pair of peak amplitudes (one from Qualisys and the other from MediaPipe Pose) that yielded the highest Pearson correlation coefficient between them.

Since each exercise was repeated multiple times during a single acquisition, segmentation was performed to extract the exercise repetitions from the recordings. Data segments corresponding to resting periods, exercise familiarization, or uncompleted trials were not considered repetitions.

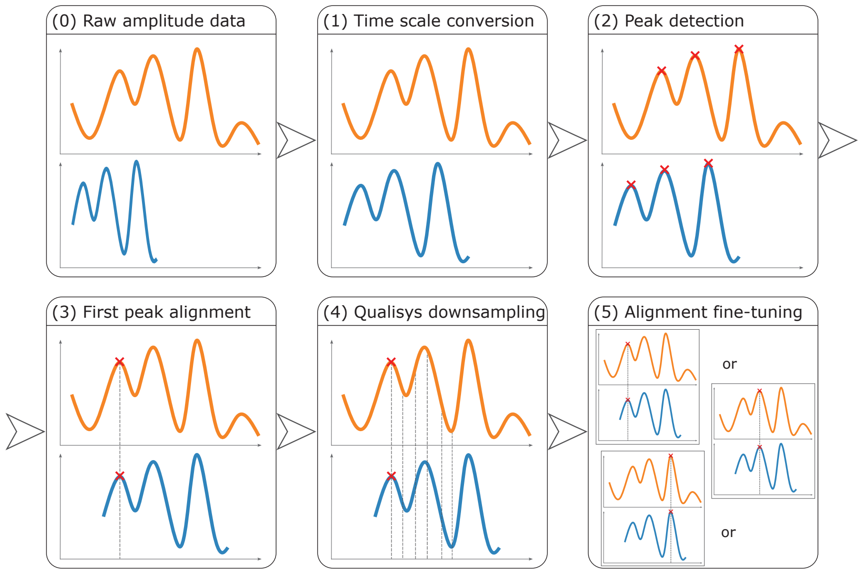


Figure 13. Data alignment between the Qualisys ground truth amplitudes (in orange) and MediaPipe Pose predicted amplitudes (in blue).

3.4. Evaluation Metrics

A comprehensive analysis of the performance of the MediaPipe Pose model in amplitude prediction was conducted by comparing the model amplitudes with the ground truth amplitudes for each exercise across all participants. Firstly, two error metrics were selected to assess the model accuracy: Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE), defined by Equations (1) and (2), respectively:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{1}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{2}$$

where n is the number of observations (i.e., the number of collected frames for each exercise), y_i is the ground truth value for observation i, and $\hat{y}_i$ is the predicted value for observation i. Then, to quantify the correlation between two variables (Qualisys ground truth amplitudes and MediaPipe Pose predicted amplitudes), two metrics were evaluated: Pearson correlation coefficient and cosine similarity. Pearson correlation coefficient measures the linear relationship between two sets of data points and takes into account both their magnitude and direction, meaning that it is sensitive to the scale of the data [32]. Additionally, cosine similarity measures similarity as the cosine of the angle between two vectors [33]. Unlike Pearson correlation, this metric is not sensitive to the magnitude of data, only to their direction, meaning that it assesses the morphology of the acquisition without considering its scale. Finally, the properties of a linear regression between the Qualisys and MediaPipe Pose data were investigated.

4. Results

For the ROM study, results consisted of pairs of ground truth amplitudes measured by Qualisys and amplitudes predicted by MediaPipe Pose collected from eight participants performing eight exercises frequently performed in physiotherapy sessions. Figure 14 shows representative traces of Qualisys ground truth (in orange) and MediaPipe Pose predicted (in blue) amplitudes over time during the SA and SKF exercises performed by Subject 1, highlighting the raw amplitude data before the alignment procedure (Figure 14a,b) and the aligned amplitude data (Figure 14c,d).

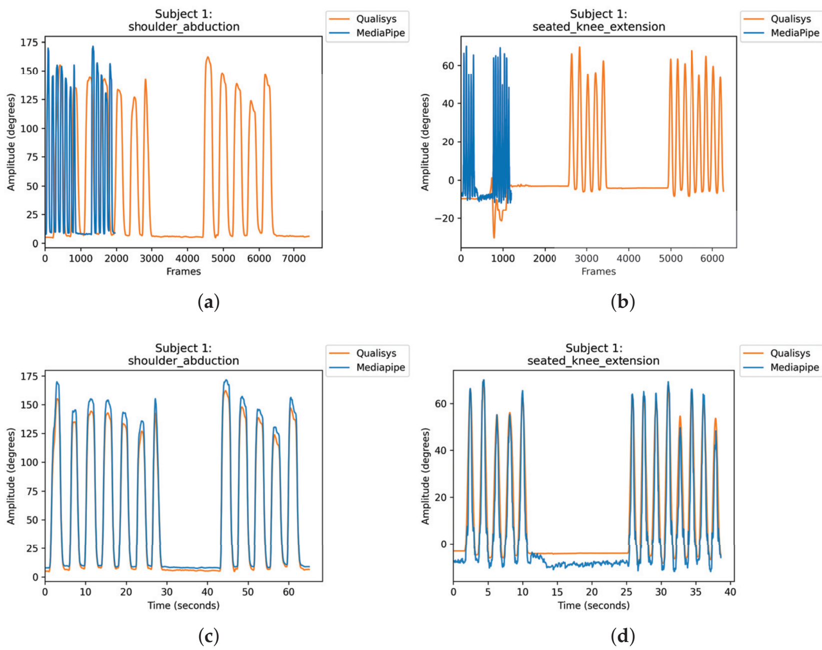


Figure 14. Example of Qualisys ground truth (in orange) and MediaPipe Pose predicted (in blue) amplitudes for Subject 1 performing SA exercise and SKF exercise. (**a**,**b**) show the raw amplitude before the alignment procedure, and (**c**,**d**) the aligned amplitude data, before segmenting the sample to extract the exercise repetitions.

The evaluation of the MediaPipe Pose model for ROM estimation was conducted from two perspectives: peak amplitudes and motion amplitudes. Peak amplitudes represent the maximum articular angle measured in each exercise repetition. It is an important parameter for physiotherapists to assess their patients' progress during treatment [7]. Moreover, investigating all amplitude predictions during exercise execution (motion amplitudes) is also important, as it can provide a comprehensive evaluation of the algorithm's performance for a wider range of angles.

MAE and MAPE (Equations (1) and (2)) for both peak and motion amplitudes are shown in Table 4. MAPE evaluation was based on previously used criteria [34], and different colors were used to highlight various performances (Table 4). For the peak amplitude analysis, MAE varied from 3.7° (HA) to 28.8° (SF), and MAPE varied from 6.6% (SKF) to 28.7% (SF). Both MAE and MAPE were higher for upper-body exercises than for lower-body exercises. For most exercises, the model had a MAPE below or close to 10% (highly accurate forecasts), suggesting promising results. For motion amplitudes, absolute angles between 0° and 1° were eliminated (threshold = 1°) to prevent infinite errors. MAE varied from 3.2° (HA) to 18.7° (SP), and MAPE between 14.9% (SA) and 107.4% (MCH). SA, SP, EF, and SQ exercises showed the lowest MAPE, while MCH, SKF, SF, and HA exercises showed the highest MAPE.

Table 4. MAE, in degrees, and MAPE, in percentage, between Qualisys and MediaPipe Pose amplitudes (peak and motion) for each exercise. MAPE color code [34]: <10% (highly accurate forecast) in green; 10–20% (good forecast) in yellow; 20–50% (reasonable forecast) in light orange; >50% (inaccurate forecast) in dark orange. (s) and (f) indicate sagittal and frontal plane exercises, respectively.

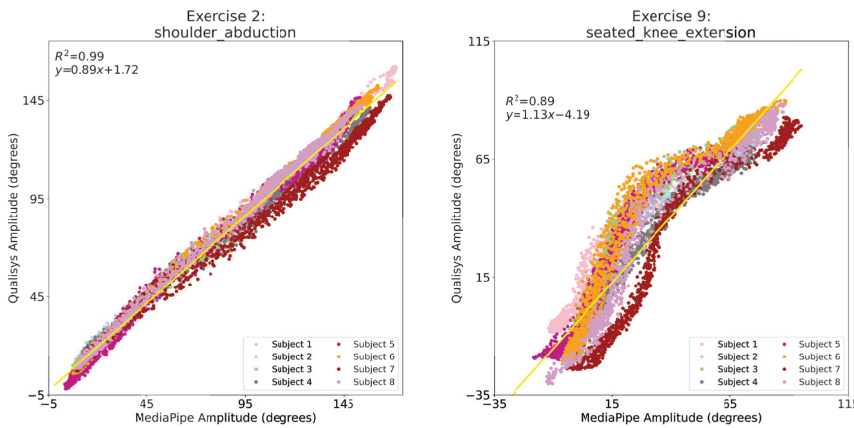
Exercise	Peak Amplitudes		Motion Amplitudes (Threshold = 1°)	
	MAE (°)	MAPE (%)	MAE (°)	MAPE (%)
1. SF (s)	28.8	28.7	15.6	66.60
2. SA (f)	13.0	10.2	7.7	14.90
3. EF (s)	11.7	9.6	10.6	24.2
4. SP (f)	13.8	9.5	18.7	23.0
5. HA (f)	3.7	9.0	3.2	62.9
6. SQ (s)	7.6	7.9	8.3	25.0
7. MCH (s)	6.3	7.7	6.3	107.4
8. SKF (s)	4.9	6.6	9.9	78.10

The results to quantify the correlation between two variables (Qualisys ground truth amplitudes and MediaPipe Pose predicted amplitudes) are shown in Table 5. For peak amplitudes, the Pearson coefficient varied between 0.744 (SP) and 0.961 (MCH). When Pearson coefficients are above 0.9 (as shown in SA, EF, HA, and MCH), the association between ground truth and predicted data is considered very strong [35]. The p-value was <0.001, indicating a statistically significant Pearson coefficient [32]. Cosine similarity was equal to or greater than 0.992 for all exercises, suggesting a strong relationship between data morphology from Qualisys and MediaPipe Pose, regardless of their magnitudes. For motion amplitudes, the Pearson coefficient was equal to or greater than 0.904 (very strong correlation [35]) for all exercises; cosine similarity values were equal to or greater than 0.940. Similarly to the MAPE study of motion amplitudes, frontal plane upper-body exercises (SA and SP), EF, and SQ exercises were the ones with the highest cosine similarity value (above 0.990).

Table 5. Correlation analysis between the Qualysis and MediaPipe Pose amplitudes (peak and motion) for each exercise: Pearson correlation coefficient (r) and cosine similarity coefficient (cos_sim). The p-value was <0.001, indicating a statistically significant Pearson coefficient. Color code: >0.9 in green; 0.8–0.9 in yellow; 0.7–0.8 in light orange. (s) and (f) indicate sagittal and frontal plane exercises, respectively.

Exercise	Peak Amplitudes		Motion Amplitudes	
	r	cos_sim	r	cos_sim
1. SF (s)	0.894	0.992	0.904	0.949
2. SA (f)	0.939	0.999	0.996	0.999
3. EF (s)	0.903	0.997	0.963	0.990
4. SP (f)	0.744	0.999	0.985	0.997
5. HA (f)	0.915	0.995	0.985	0.987
6. SQ (s)	0.833	0.998	0.981	0.993
7. MCH (s)	0.961	0.996	0.964	0.979
8. SKF (s)	0.765	0.997	0.942	0.961

Amplitudes measured by Qualisys and predicted by MediaPipe Pose were displayed on plots to visualize the relationship between ground truth and predictions, demonstrated by the high Pearson coefficient and cosine similarity seen in Table 5. Representative plots of the relationship between Qualisys and MediaPipe Pose motion amplitudes for SA and SKF exercises are shown in Figure 15. Results for all exercises are in Table 6. The coefficient of determination (R^2) assesses how well the linear regression fits the data. R^2 values ranged between 0.82 (SF) and 0.99 (SA). R^2 values were the highest for the frontal plane (SA, SP, HA) and SQ exercises. MCH and EF also showed high R^2, while SKF and SF were the exercises with the lowest R^2 results. Frontal plane exercises, SQ, and MCH plots showed an approximately linear relationship between data (Figure 15a), and SF, SKF, and EF showed an approximately cubic polynomial relationship between data (Figure 15b).



(**a**) (**b**)

Figure 15. Relation between Qualisys and MediaPipe Pose motion amplitudes for (**a**) SA exercise and (**b**) SKF exercise. Each color represents a different subject, and the yellow line is the linear regression that best fits the amplitude data for the exercise; the coefficient of determination (R^2) and the linear regression equation (slope and intercept) are also shown, where y and x are the Qualisys and MediaPipe Pose amplitudes, respectively.

Table 6. Linear regression between motion amplitudes of Qualisys and MediaPipe Pose for each exercise: slope and intercept values for the equation that better fits the transformation of predictions (MediaPipe Pose points) into expected data (Qualisys points), coefficient of determination (R^2), and curve shape. (s) and (f) indicate sagittal and frontal plane exercises, respectively.

Exercise	Motion Amplitudes			
	Slope	**Intercept**	R^2	**Curve Shape**
1. SF (s)	0.75	11.3	0.82	Not linear
2. SA (f)	0.89	1.72	0.99	Linear
3. EF (s)	1.23	−11.86	0.93	Not linear
4. SP (f)	0.96	−14.2	0.97	Linear
5. HA (f)	0.92	3.39	0.97	Linear
6. SQ (s)	1.05	5.12	0.96	Linear
7. MCH (s)	1.03	−0.82	0.93	Linear
8. SKF (s)	1.13	−4.19	0.89	Not linear

5. Discussion

Error and correlation analyses provided valuable insights about the MediaPipe Pose performance. Firstly, the model predictions for peak amplitudes showed low MAE and MAPE, indicating a promising performance of MediaPipe Pose, in particular for lower-body exercises, where the ROM is commonly lower than in upper-body exercises. Secondly, the model predictions for motion amplitudes showed higher MAE and MAPE than for peak amplitudes, suggesting that the algorithm performed better in static poses (peak amplitudes) than in dynamic movements (motion amplitudes) [35]; furthermore, MediaPipe Pose showed lower MAPE for frontal plane upper-body (SA and SP), EF, and SQ exercises than for HA, SF, MCH, and SKF exercises.

SA is a frontal plane, upper-body, and unilateral exercise. The model's performance for frontal plane exercises was expected to be better than for sagittal plane exercises since, for the frontal configuration, the movement occurs in a plane approximately parallel to the camera plane; thus, all joints are approximately at the same angle to the camera (same depth). Depth ambiguities, mainly associated with side-views where different depths need to be determined for each joint, are a primary challenge in monocular 3D HPE [36], which are avoided in frontal plane exercises. Besides depth ambiguities, other factors influence the 3D estimations, justifying why other frontal plane exercises (SP and HA) did not show performance as good as the SA exercise. For instance, training data with a lack of representative examples for movements, such as SP and HA, may also make the prediction by the algorithm more challenging [37].

For motion amplitudes, EF is a sagittal plane, upper-body, and unilateral exercise with one of the lowest MAPE. In this exercise, only the right forearm (i.e., right wrist joint) was moving, and during the exercise execution, this body segment was not occluded. The low MAPE values for motion amplitudes suggested that MediaPipe Pose can correctly predict the right elbow and wrist (only joints considered for this exercise). On the other hand, SKF was the exercise with one of the highest MAPE. This exercise involves unusual poses and self-occlusions that may be challenging for MediaPipe Pose. This is likely due to the lack of training data representing those poses and self-occlusions [17,37].

SQ and MCH exercises are more prone to self-occlusions and unusual poses since their execution involves moving several joints simultaneously. Interestingly, SQ was one of the exercises with the lowest amplitude error (MAPE), probably because the right knee, hip, and ankle are the only joints evaluated in the ROM analysis and these are visible (not occluded) during the exercise execution. The MCH exercise is a bilateral movement, where each leg moves one at a time. Although only the right leg (further from the camera) motion was evaluated, the motion of the left leg (closer to the camera) in front of the right leg (self-occlusion) may contribute to a high amplitude error (MAPE) [17].

In summary, MediaPipe Pose errors reported previously may be due to depth ambiguities [38], self-occlusions [17], or challenging poses [39]. Furthermore, erroneous 2D estimations may also affect the 3D HPE [40]; MediaPipe Pose estimates the 3D positions from the predicted 2D positions (*2D to 3D lifting* technique). 3D HPE models may incorporate *2D to 3D lifting* techniques, where an intermediate 2D pose is first estimated, and then lifted to 3D, i.e., estimate the 3D position of the joints from their 2D positions, meaning that higher 2D errors may contribute to higher 3D errors. The amount of annotated data can also influence deep learning algorithms; wider variability of scenarios (unusual poses and occlusions) present in the training dataset has been found to contribute to a better performance of these models [41]. The MediaPipe Pose model was trained on a customized dataset [23,29], capturing a wide range of fitness poses. Nevertheless, some poses from the selected exercises may not be widely represented in its dataset, making the model prediction harder. Additionally, the experimental data acquisition by the Qualisys system [42,43], as well as data preprocessing oversimplification when converting 3D joint positions into amplitude values, may introduce errors in the ground truth data.

For motion amplitudes, despite the high MAPE results seen in some exercises (as high as 107.4%), a strong correlation between Qualisys ground truth and MediaPipe Pose predicted amplitudes was observed (Pearson coefficient and cosine similarity above 0.9) [35]. Taken together, MAPE and correlation results seemed to indicate that MediaPipe Pose predictions replicated the shape of the curve of the Qualisys data (high correlation), but shifted or scaled (high MAPE). Therefore, for the frontal plane, SQ and MCH exercises, the linear function shown in Table 6 could be used to map MediaPipe Pose predictions into Qualisys ground truth amplitudes, in order to decrease the error between them. Similarly, a fine-tuned cubic polynomial function could be applied to SF, EF, and SKF exercises [44].

Overall, the exercise with the best results was SA, a frontal plane, upper-body, and unilateral exercise, while the exercises with the lowest performance were the MCH and SKF, where complex poses and numerous occlusions can be found, and SF, where the depth may be harder to estimate.

6. Conclusions

This work explored the potential of using approaches for 3D HPE from monocular 2D video feeds in musculoskeletal telerehabilitation. Specifically, the performance of MediaPipe Pose was evaluated on a wide range of exercises commonly performed in physiotherapy sessions, covering different body poses. The investigation of MediaPipe Pose for 3D HPE was focused on joint ROM, the metric used by physiotherapists in both telerehabilitation and conventional sessions to follow patients. As a valuable addition to existing research, this study also provided an in-depth description of (1) a novel approach for defining a 3D Cartesian coordinate system, invariant to camera orientation, suitable for both 2D and 3D camera acquisitions; (2) the calculation of the Range of Motion (ROM), considering the physiotherapy definition; and (3) the description of the data alignment procedure for acquisition systems with different frame rates.

The MediaPipe Pose model yielded promising results, with the SA, SP, EF, and SQ exercises generally showing better performance than the remaining exercises. Overall, the exercise with the best results was SA, a frontal plane, upper-body, and unilateral exercise, while the exercises with the lowest performance were the MCH, SKF, and SF.

In conclusion, this study supports the potential of using MediaPipe Pose for 3D body tracking from monocular 2D videos in musculoskeletal telerehabilitation applications, to eliminate the need for complex specialized hardware, such as 3D depth cameras or wearables. Although the results varied under different conditions, the MediaPipe Pose model showed encouraging performance. Future work should include testing other state-of-the-art algorithms, increasing the sample size of participants in the study, extending the dataset to subjects with musculoskeletal diseases, investigating post-preprocessing techniques to enhance the results, and gathering additional training data focused on physiotherapy-specific motions (such as the Fit3D dataset [45]) and poses to handle challenges, such as occlusions and depth ambiguities. With further refinements to the models, 3D body tracking from monocular 2D video feeds appears to be a viable, affordable, and accessible approach for musculoskeletal telerehabilitation solutions. In the future, this can help the development of better physiotherapy options for patients with musculoskeletal disorders.

Author Contributions: Conceptualization, H.P.d.S., G.C. and J.F.P.; methodology, C.C., H.P.d.S., D.C.F.S., A.M.M. and G.C.; software, data curation, writing—original draft preparation, and visualization, C.C.; validation, H.P.d.S.; formal analysis, C.C. and H.P.d.S.; investigation, C.C., D.C.F.S. and A.M.M.; writing—review and editing, all authors; supervision, resources and project administration, H.P.d.S., D.C.F.S., A.M.M., G.C. and J.F.P.; funding acquisition, H.P.d.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by the Fundação para a Ciência e Tecnologia (FCT)/Ministério da Ciência, Tecnologia e Ensino Superior (MCTES) through national funds and when applicable co-funded by EU funds under the projects UIDB/50008/2020, and Scientific Employment Stimulus—Individual Call—2022.04901.CEECIND.

Institutional Review Board Statement: The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Ethics Committee of the Escola Superior de Saúde de Santa Maria (Reference: CE2022/09 approved on the 18 May 2022).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data and source code underlying this research are openly available. The dataset of the 3D positions measured by the Qualysis Motion Capture system and the dataset of the 3D positions predicted by the MediaPipe Pose model are available at the following Zenodo repository: https://zenodo.org/records/10408307. The source code is available at the following GitHub repository: https://github.com/carolinaclemente00/3D-HPE-MediaPipe-Pose/.

Acknowledgments: The authors would like to thank all the volunteers that have agreed to participate in the study, and all the institutions that made this work possible.

Conflicts of Interest: The authors declare no conflicts of interest. The sponsors had no role in the design, execution, interpretation, or writing of the study.

Abbreviations

The following abbreviations are used in this manuscript:

HPE	Human Pose Estimation
MoCap	Motion Capture
ROM	Range of Motion
FPS	Frames per Second
SF	Shoulder Flexion/Extension
SA	Shoulder Abduction/Adduction
EF	Elbow Flexion/Extension
SP	Shoulder Press
HA	Hip Abduction/Adduction
SQ	Squat
MCH	March
SKF	Seated Knee Flexion/Extension
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error

References

1. Cieza, A.; Causey, K.; Kamenov, K.; Hanson, S.W.; Chatterji, S.; Vos, T. Global estimates of the need for rehabilitation based on the Global Burden of Disease study 2019: A systematic analysis for the Global Burden of Disease Study 2019. *Lancet* **2020**, *396*, 2006–2017. [CrossRef] [PubMed]
2. Vieira, L.M.S.M.d.A.; de Andrade, M.A.; Sato, T.d.O. Telerehabilitation for musculoskeletal pain–An overview of systematic reviews. *Digit. Health* **2023**, *9*, 20552076231164242. [CrossRef] [PubMed]
3. Cottrell, M.A.; Russell, T.G. Telehealth for musculoskeletal physiotherapy. *Musculoskelet. Sci. Pract.* **2020**, *48*, 102193. [CrossRef]
4. Areias, A.C.; Costa, F.; Janela, D.; Molinos, M.; Moulder, R.G.; Lains, J.; Scheer, J.K.; Bento, V.; Yanamadala, V.; Correia, F.D. Long-term clinical outcomes of a remote digital musculoskeletal program: An ad hoc analysis from a longitudinal study with a non-participant comparison group. *Healthcare* **2022**, *10*, 2349. [CrossRef] [PubMed]
5. Dias, G.; Adrião, M.L.; Clemente, P.; da Silva, H.P.; Chambel, G.; Pinto, J.F. Effectiveness of a Gamified and Home-Based Approach for Upper-limb Rehabilitation. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Scotland, UK, 11–15 July 2022; pp. 2602–2605.
6. Khan, M.H.; Zöller, M.; Farid, M.S.; Grzegorzek, M. Marker-based movement analysis of human body parts in therapeutic procedure. *Sensors* **2020**, *20*, 3312. [CrossRef] [PubMed]
7. Clarkson, H.M. *Musculoskeletal Assessment-Joint Motion and Muscle Testing*; Lippincott Williams & Wilkins: Philadelphia, PA, USA, 2013.
8. Alarcón-Aldana, A.C.; Callejas-Cuervo, M.; Bo, A.P.L. Upper limb physical rehabilitation using serious videogames and motion capture systems: A systematic review. *Sensors* **2020**, *20*, 5989. [CrossRef] [PubMed]
9. Correia, F.D.; Nogueira, A.; Magalhães, I.; Guimarães, J.; Moreira, M.; Barradas, I.; Molinos, M.; Teixeira, L.; Pires, J.; Seabra, R.; et al. Digital versus conventional rehabilitation after total hip arthroplasty: A single-center, parallel-group pilot study. *JMIR Rehabil. Assist. Technol.* **2019**, *6*, e14523. [CrossRef] [PubMed]
10. Correia, F.D.; Nogueira, A.; Magalhães, I.; Guimarães, J.; Moreira, M.; Barradas, I.; Molinos, M.; Teixeira, L.; Tulha, J.; Seabra, R.; et al. Medium-term outcomes of digital versus conventional home-based rehabilitation after total knee arthroplasty: Prospective, parallel-group feasibility study. *Jmir Rehabil. Assist. Technol.* **2019**, *6*, e13111. [CrossRef]
11. Pinto, J.F.; Carvalho, H.R.; Chambel, G.R.; Ramiro, J.; Goncalves, A. Adaptive gameplay and difficulty adjustment in a gamified upper-limb rehabilitation. In Proceedings of the IEEE International Conference on Serious Games and Applications for Health (SeGAH), Vienna, Austria, 16–18 May 2018; pp. 1–8.

12. Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N.; Shah, M. Deep learning-based human pose estimation: A survey. *ACM Comput. Surv.* **2023**, *56*, 1–37. [CrossRef]
13. Milosevic, B.; Leardini, A.; Farella, E. Kinect and wearable inertial sensors for motor rehabilitation programs at home: State of the art and an experimental comparison. *Biomed. Eng. Online* **2020**, *19*, 1–26. [CrossRef]
14. Colyer, S.L.; Evans, M.; Cosker, D.P.; Salo, A.I. A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system. *Sport Med.-Open* **2018**, *4*, 1–15. [CrossRef] [PubMed]
15. Kadambi, A.; Bhandari, A.; Raskar, R. 3D Depth Cameras in Vision: Benefits and Limitations of the Hardware: With an Emphasis on the First-and Second-Generation Kinect Models. *Comput. Vis. Mach. Learn. RGB-D Sens.* **2014**, 3–26. [CrossRef]
16. Hellsten, T.; Karlsson, J.; Shamsuzzaman, M.; Pulkkis, G. The Potential of Computer Vision-Based Markerless Human Motion Analysis for Rehabilitation. *Rehabil. Process. Outcome* **2021**, *10*, 11795727211022330. [CrossRef] [PubMed]
17. Cheng, Y.; Yang, B.; Wang, B.; Yan, W.; Tan, R.T. Occlusion-aware networks for 3D human pose estimation in video. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 723–732.
18. Mehta, D.; Sotnychenko, O.; Mueller, F.; Xu, W.; Elgharib, M.; Fua, P.; Seidel, H.; Rhodin, H.; Pons-Moll, G.; Theobalt, C. XNect: Real-time multi-person 3D motion capture with a single RGB camera. *ACM Trans. Graph. (TOG)* **2020**, *39*, 82:1–82:17. [CrossRef]
19. Kendall, A.; Grimes, M.; Cipolla, R. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2938–2946.
20. Bajpai, R.; Joshi, D. MoveNet: A deep neural network for joint profile prediction across variable walking speeds and slopes. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–11.
21. Fang, H.S.; Li, J.; Tang, H.; Xu, C.; Zhu, H.; Xiu, Y.; Li, Y.L.; Lu, C. AlphaPose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 7157–7173. [CrossRef] [PubMed]
22. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
23. Bazarevsky, V.; Grishchenko, I.; Raveendran, K.; Zhu, T.; Zhang, F.; Grundmann, M. BlazePose: On-device real-time body pose tracking. *arXiv* **2020**, arXiv:2006.10204.
24. Bolaños, C.; Fernández-Bermejo, J.; Dorado, J.; Agustín, H.; Villanueva, F.J.; Santofimia, M.J. A comparative analysis of pose estimation models as enablers for a smart-mirror physical rehabilitation system. *Procedia Comput. Sci.* **2022**, *207*, 2536–2545. [CrossRef]
25. Huang, Y.P.; Chou, Y.J.; Lee, S.H. An OpenPose-based System for Evaluating Rehabilitation Actions in Parkinson's Disease. In Proceedings of the International Automatic Control Conference (CACS), Kaohsiung, Taiwan, 3–6 November 2022; pp. 1–6.
26. Lv, X.; Ta, N.; Chen, T.; Zhao, J.; Wei, H. Analysis of Gait Characteristics of Patients with Knee Arthritis Based on Human Posture Estimation. *Biomed Res. Int.* **2022**, *2022*, 7020804. [CrossRef]
27. Marusic, A.; Nguyen, S.M.; Tapus, A. Evaluating Kinect, OpenPose and BlazePose for Human Body Movement Analysis on a Low Back Pain Physical Rehabilitation Dataset. In Proceedings of the Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction, Stockholm, Sweden, 13–16 March 2023; pp. 587–591.
28. Mehta, D.; Sridhar, S.; Sotnychenko, O.; Rhodin, H.; Shafiei, M.; Seidel, H.P.; Xu, W.; Casas, D.; Theobalt, C. Vnect: Real-time 3D human pose estimation with a single RGB camera. *ACM Trans. Graph. (TOG)* **2017**, *36*, 1–14. [CrossRef]
29. Xu, H.; Bazavan, E.G.; Zanfir, A.; Freeman, W.T.; Sukthankar, R.; Sminchisescu, C. Ghum & Ghuml: Generative 3D human shape and articulated pose models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6184–6193.
30. Ionescu, C.; Papava, D.; Olaru, V.; Sminchisescu, C. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 1325–1339. [CrossRef] [PubMed]
31. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
32. Asuero, A.G.; Sayago, A.; González, A. The correlation coefficient: An overview. *Crit. Rev. Anal. Chem.* **2006**, *36*, 41–59. [CrossRef]
33. Xia, P.; Zhang, L.; Li, F. Learning similarity with cosine similarity ensemble. *Inf. Sci.* **2015**, *307*, 39–52. [CrossRef]
34. Lewis, C.D. *Industrial and Business Forecasting Methods: A Practical Guide to Exponential Smoothing and Curve Fitting*; Butterworth Scientific: Boston, MA, USA, 1982. [CrossRef]
35. Slowik, J.S.; McCutcheon, T.W.; Lerch, B.G.; Fleisig, G.S. Comparison of a Single-view Image-based System to a Multi-camera Marker-based System for Human Static Pose Estimation. *J. Biomech.* **2023**, *159*, 111746. [CrossRef]
36. Zhang, S.; Wang, C.; Dong, W.; Fan, B. A survey on depth ambiguity of 3D human pose estimation. *Appl. Sci.* **2022**, *12*, 10591. [CrossRef]
37. Qiu, Z.; Qiu, K.; Fu, J.; Fu, D. Weakly-supervised pre-training for 3D human pose estimation via perspective knowledge. *Pattern Recognit.* **2023**, *139*, 109497. [CrossRef]
38. Zhang, D.; Wu, Y.; Guo, M.; Chen, Y. Deep learning methods for 3D human pose estimation under different supervision paradigms: A survey. *Electronics* **2021**, *10*, 2267. [CrossRef]
39. Liu, Z.; Zhu, J.; Bu, J.; Chen, C. A survey of human pose estimation: The body parts parsing based methods. *J. Vis. Commun. Image Represent.* **2015**, *32*, 10–19. [CrossRef]

40. Hossain, M.R.I. Understanding the Sources of Error for 3D Human Pose Estimation from Monocular Images and Videos. Ph.D. Thesis, University of British Columbia, Kelowna, BC, Canada, 2017.
41. Andriluka, M.; Pishchulin, L.; Gehler, P.; Schiele, B. 2D human pose estimation: New benchmark and state of the art analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA 23–28 June 2014; pp. 3686–3693.
42. Barré, A.; Thiran, J.P.; Jolles, B.M.; Theumann, N.; Aminian, K. Soft tissue artifact assessment during treadmill walking in subjects with total knee arthroplasty. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 3131–3140. [CrossRef]
43. Moro, M.; Marchesi, G.; Hesse, F.; Odone, F.; Casadio, M. Markerless vs. marker-based gait analysis: A proof of concept study. *Sensors* **2022**, *22*, 2011. [CrossRef] [PubMed]
44. Seber, G.A.; Lee, A.J. Polynomial regression. In *Linear Regression Analysis*; Wiley Series in Probability and Statistics; John Wiley & Sons: Hoboken, NJ, USA, 2003; pp. 165–185. [CrossRef]
45. Fieraru, M.; Zanfir, M.; Pirlea, S.C.; Olaru, V.; Sminchisescu, C. AIFit: Automatic 3D Human-Interpretable Feedback Models for Fitness Training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 9919–9928.

sensors

MDPI

Article

Reducing the Impact of Sensor Orientation Variability in Human Activity Recognition Using a Consistent Reference System

Manuel Gil-Martín *, Javier López-Iniesta , Fernando Fernández-Martínez and Rubén San-Segundo

Speech Technology and Machine Learning, Information Processing and Telecommunications Center, E.T.S.I. de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain; javier.lopeziniesta.diazdelcampo@alumnos.upm.es (J.L.-I.); fernando.fernandezm@upm.es (F.F.-M.)
* Correspondence: manuel.gilmartin@upm.es

Abstract: Sensor- orientation is a critical aspect in a Human Activity Recognition (HAR) system based on tri-axial signals (such as accelerations); different sensors orientations introduce important errors in the activity recognition process. This paper proposes a new preprocessing module to reduce the negative impact of sensor-orientation variability in HAR. Firstly, this module estimates a consistent reference system; then, the tri-axial signals recorded from sensors with different orientations are transformed into this consistent reference system. This new preprocessing has been evaluated to mitigate the effect of different sensor orientations on the classification accuracy in several state-of-the-art HAR systems. The experiments were carried out using a subject-wise cross-validation methodology over six different datasets, including movements and postures. This new preprocessing module provided robust HAR performance even when sudden sensor orientation changes were included during data collection in the six different datasets. As an example, for the WISDM dataset, sensors with different orientations provoked a significant reduction in the classification accuracy of the state-of-the-art system (from 91.57 $\pm$ 0.23% to 89.19 $\pm$ 0.26%). This important reduction was recovered with the proposed algorithm, increasing the accuracy to 91.46 $\pm$ 0.30%, i.e., the same result obtained when all sensors had the same orientation.

Keywords: human activity recognition; gravity estimation; sensor-orientation-independent; forward movement direction; wearable sensors; acceleration signals; deep learning; convolutional neural networks

Citation: Gil-Martín, M.; López-Iniesta, J.; Fernández-Martínez, F.; San-Segundo, R. Reducing the Impact of Sensor Orientation Variability in Human Activity Recognition Using a Consistent Reference System. *Sensors* **2023**, *23*, 5845. https://doi.org/10.3390/s23135845

Academic Editors: Antonio Fernández-Caballero and Michael E. Hahn

Received: 18 May 2023
Revised: 11 June 2023
Accepted: 22 June 2023
Published: 23 June 2023

1. Introduction

In the last decade, there has been an increasing interest in Human Activity Recognition (HAR) for recognizing the physical activities that people perform during their day-to-day lives. State-of-the-art systems for HAR integrate different signal processing and deep learning techniques [1–8], reaching promising results in different applications, e.g., sports monitoring [9–11] (such as fitness tracking, training adaptation, or personal incentivizing), rehabilitation [12], and respiratory diseases spread minimization [13].

In these applications, sensor orientation is a critical aspect when using tri-axial signals (such as accelerations); different sensor orientations can introduce important errors in the activity recognition process. In real scenarios (not supervised by experts), people place their sensors in different orientations, compromising the system performance. For example, fit bands and smartwatches can rotate along the wrist, presenting very different orientations. This fact affects the final recognition performance, requiring strategies to reduce the negative impact of these changes.

State-of-the-art HAR systems include robust features for dealing with different sensor orientations, but they do not incorporate specific algorithms for correcting the mismatches in sensor orientation. This work proposes a preprocessing module to correct these mismatches by extracting a consistent reference system and transforming the tri-axial acceleration signals from the sensor reference system to this consistent reference system. The proposed algorithm increases the robustness of the HAR system against sensor orientation changes, obtaining important improvements over several state-of-the-art systems and datasets. The main contributions of this paper are as follows:

- A new preprocessing algorithm for mitigating the effects of sensor orientation variability. Firstly, this algorithm generates a consistent reference system from the estimation of gravitational and forward movement directions. Secondly, the tri-axial acceleration recorded from the sensor is transformed from the sensor reference system to the consistent reference system. This proposal has demonstrated robust activity recognition even when sudden and abrupt sensor orientation changes happened during data recording.
- A study of the effect of the proposed algorithm depending on the type of activity, i.e., movements or postures.
- The evaluation of the proposal using six well-known HAR systems and datasets in a subject-wise cross-validation scenario, including a wide variety of subjects, activities, devices, and locations.

This paper is organized as follows. Section 2 reviews the literature, discussing different previous works related to the topic of this work. Section 3 describes the materials and methods used, including the system architecture, the proposed algorithm for adapting the sensor orientation through a consistent reference system, the signal processing and deep learning approaches, and the evaluation details. Section 4 describes the datasets and discusses the experiments and the obtained results. Finally, Section 5 summarizes the main conclusions of the paper.

2. Related Works

In the HAR field, the use of wearable devices is widely extended [14–16]. HAR systems based on tri-axial inertial signals (such as accelerations) have an important problem regarding the sensor orientations. In real scenarios (not supervised by experts), people place their sensors in different orientations (fit bands and smartwatches can rotate along the wrist), compromising the system performance.

In the literature, there are few proposals dealing with different sensor orientations [17]. Most of these studies have focused their contributions on extracting robust features (such as the magnitude of the accelerometer vector [18] or specific vertical and horizontal features [19]) that are not sensitive to orientation changes. These features showed good robustness, but they did not obtain the best performance. In the same way, San-Segundo et al. [20] used different mitigation techniques to deal with heterogeneities in HAR using smartphones and smartwatches. They used different feature extraction strategies such as filtering or session-specific normalization of feature data.

Only two previous works have been identified which propose similar algorithms to compute consistent axes for representing tri-axial acceleration. In the first study, Henpraserttae et al. [21] estimated the vertical and forward–backward axes of the body in several movements. These initial experiments were very promising, but the experimental setup only included recordings from five subjects and six daily activities, using data from the same subjects to train and test their system. In a real application, the system must be tested with different subjects compared to those used in training. The second work [22] developed two methods to remove the effect of absolute sensor orientation from the raw sensor data: a heuristic transformation and a singular value decomposition-based transformation. These techniques did not provide significant improvements for reducing the degradation due to wrong sensor orientation.

This paper solves the limitations of these previous works, proposing a new algorithm for estimating a full consistent reference system (composed of three main axes) and transforming tri-axial acceleration from the sensor system to the consistent reference system. This preprocessing algorithm has been evaluated on state-of-the-art HAR systems considering six different datasets in realistic scenarios (training and testing with different subjects). This work covers the research gap between these preliminary studies and the proposal of a full solution evaluated with several datasets in realistic scenarios.

In order to complete the related work section, it is interesting to comment that the analysis of acceleration signals has been applied to other fields such as driving [23]. In this work, deep learning techniques have been used to estimate the vehicle movement direction. This forward movement directional vector is useful for multiple applications such as characterizing driving styles or detecting dangerous events. In addition, some previous works [24–26] have performed IMU-based attitude estimation in automotive and human–robotic interaction systems. In fact, the authors have analyzed the effects of the bias and error of the IMUs and introduce other signals to improve the attitude estimation. Regarding driving applications, there also exist previous works [27] that have been focused on vehicle trajectory extraction, reconstruction, and evaluation to develop automated driving systems.

3. Materials and Methods

This section includes information about the proposed algorithm to adapt the acceleration coordinates to a consistent reference system, the architecture of the state-of-the-art system (describing the main modules), and the evaluation of the systems in different scenarios.

3.1. System Architecture

Figure 1 shows a general module diagram of an HAR system, incorporating the new algorithm as a previous step before the signal processing. Once the acceleration data are obtained from inertial units or smartphones, the consistent reference system is obtained. Then, the original acceleration is transformed from the sensor coordinates to this consistent reference system. This new representation is independent of the sensor orientation. Afterward, it is possible to apply signal processing techniques such as Fast Fourier Transform to extract relevant information from the motion signals in the frequency domain. Finally, these spectra are included in a Convolutional Neural Network to model and classify the different physical activities. This architecture has been used to obtain state-of-the-art performance in the datasets considered in this study.

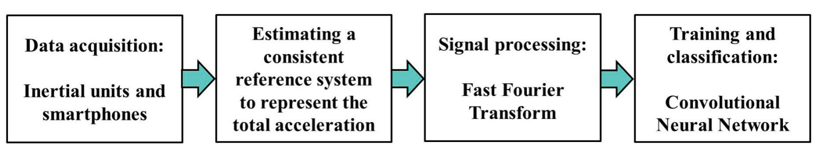


Figure 1. System architecture including the new algorithm before signal processing.

3.2. Estimating a Consistent Reference System to Represent the Total Acceleration

The proposed approach aims to create a consistent reference system independently of the sensor orientation and represent the total acceleration according to this consistent reference system. This process is composed of the following main steps (see Figure 2):

1. Firstly, the gravity vector is estimated from the total acceleration recorded from the accelerometer. Each coordinate of gravity is computed by applying a sliding mean over the three coordinates (X, Y, and Z in the sensor reference system) of the total acceleration through a convolution operation. Computing the average, we remove subject movements, leaving only the gravity [28]. For this step, a sliding mean filter of 5 s was used to compute the mean over each gravitational coordinate. In this way, we obtain the three components of gravity at each sample point. The filter size was analyzed in preliminary experiments, but it did not affect the results.
2. Secondly, we obtain the horizontal acceleration by subtracting the component in the gravity (vertical) direction from the total acceleration vector. After subtracting

the vertical component, we compute the forward direction at each sample point by applying a sliding mean (5 s) over the horizontal acceleration. Unit vectors in the gravitational and the forward directions are computed dividing the original vectors by their magnitudes.

3. Thirdly, to complete the three axes system, the third unit vector is computed (at each sample point) as the cross product of gravitational and forward unit vectors.
4. Finally, the algorithm computes the new coordinates of the total acceleration according to this new reference system. The transformation of the total acceleration coordinates from the sensor reference system to the consistent reference system is accomplished by using Equation (1), where the sub-index "new" denotes the consistent reference system, and the sub-index "orig" refers to the sensor reference system. For example, u_{x_new} refers to the x coordinate of the new reference system, while u_{x_orig} refers to the x coordinate of the sensor reference system. In this equation, V_{x_new}, V_{y_new}, and V_{z_new} are the acceleration coordinates according to the consistent reference system, and V_{x_orig}, V_{y_orig} and, V_{z_orig} are the acceleration coordinates respect to the sensor reference system. The unit vectors of the consistent reference system are the forward (x), gravitational (y), and cross-based computed direction (z) vectors. To transform the acceleration from one reference system to another, it is necessary to use the three coordinates of the unitary vector for both sensor and consistent reference systems. These coordinates are referred to the sensor reference system. These unitary vectors (and their coordinates) are used to compute the elements of the transformation matrix as shown in Equation (1).

$$\begin{bmatrix} V_{x\_new} \\ V_{y\_new} \\ V_{z\_new} \end{bmatrix} = \begin{bmatrix} u_{x\_new} \cdot u_{x\_orig} & u_{x\_new} \cdot u_{y\_orig} & u_{x\_new} \cdot u_{z\_orig} \\ u_{y\_new} \cdot u_{x\_orig} & u_{y\_new} \cdot u_{y\_orig} & u_{y\_new} \cdot u_{z\_orig} \\ u_{z\_new} \cdot u_{x\_orig} & u_{z\_new} \cdot u_{y\_orig} & u_{z\_new} \cdot u_{z\_orig} \end{bmatrix} \begin{bmatrix} V_{x\_orig} \\ V_{y\_orig} \\ V_{z\_orig} \end{bmatrix} \quad (1)$$

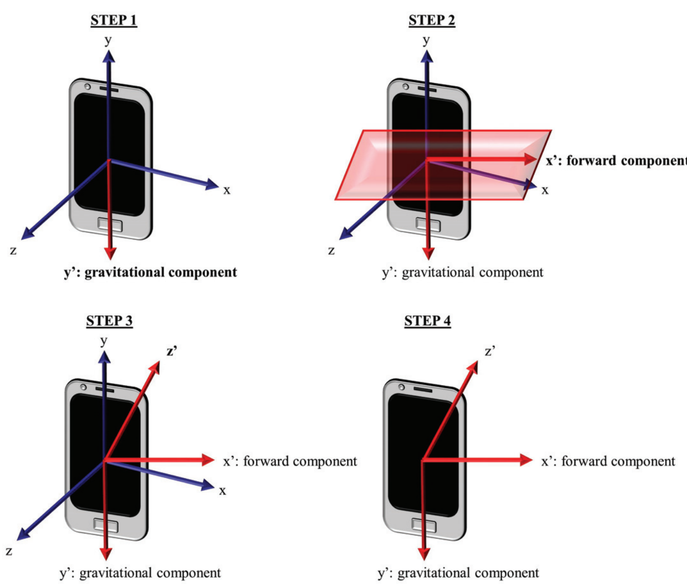


Figure 2. Generation of the consistent reference system through the algorithm. (Red part of Step 2 Maybe you could say that it represents the plane perpendicular to gravitational component that contains forward component.)

Figure 2 shows the evolution of the algorithm, including the representation of the motion vector based on the sensor reference system and the generation of the consistent reference system.

This proposed approach computes the forward-direction vector of the acceleration signal to build the consistent reference system. The computation of the consistent reference system requires an additional computational overhead of 10%, but it does not affect the real time nature of the system. Even this additional computation overhead is reached when including the computation of the consistent reference, the real-time inference time was RT < 10^{-2} (1 h of signal is inference in of than 10 s).

This algorithm works for movements such as running or cycling, but it has an important limitation when dealing with posture classification. When a person is, for example, sitting or standing, there is no motion in forward direction (this vector is zero), so it is not possible to compute any consistent reference system. To solve this limitation, we have used the gravitational component as a reference: in the case of postures, instead of using the described algorithm, we directly subtracted the component of the gravity direction from the total acceleration signals of the postures. The gravity is computed from movements, not from postures, so it was necessary to have an initial module for separating postures and motion activities as we will see later.

3.3. Signal Processing and Deep Learning Approaches

After the new preprocessing, a signal processing module divides the recording signals into analysis windows and compute the spectrum of each window. The posterior classification module aims to identify the activity at each window. The system segments the physical activity recordings using overlapped Hanning windows of 5 s using a step of 1 s. This configuration has been successfully used in previous works [2,29,30] reporting state-of-the-art performance.

After windowing, we compute the Fast Fourier Transform to generate the spectrum of each window in a range of frequencies between 0 and 10 or 20 Hz depending on the sampling frequency of the dataset. The magnitude bins of the spectrum are the input to the deep neural network. The use of FFT coefficients is justified because these features offered better results than time domain sequences or time domain features in previous works [20,31].

In state-of-the-art HAR systems, the recognition module is based on deep learning algorithms such as CNN [3,29]. The best deep learning architecture proposed in previous works [2,29], and also used in this work, is composed of two main parts: a feature learning subnet and a classification subnet. The first subnet learns features from window spectra, using two convolutional layers with intermediate max pooling layers. The second subnet uses fully connected layers to classify the learned features as a predicted activity. The architecture includes layers after max pooling and fully connected layers to avoid overfitting during training. The last layer uses a SoftMax activation function to offer the predictions of each activity for every analysis window, while intermediate layers use ReLU for reducing the impact of the gradient vanishing effect. We use categorical cross-entropy as the loss metric and the root-mean-square propagation method as the optimizer. This deep neural network has been used to model the data from all the analyzed datasets, using a different number of neurons in the last layer depending on the number of classes in each dataset.

Figure 3 represents the architecture used in this work to model and classify the physical activity of all the datasets, where M denotes the number of samples for each signal (corresponding to the magnitude bins of the spectra) and C indicates the number of recognized activities.

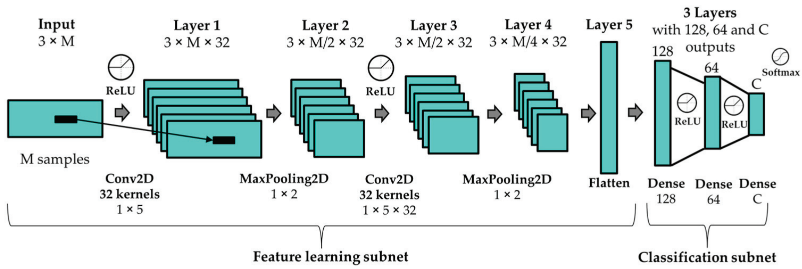


Figure 3. Convolutional Neural Network architecture used in this work for all the datasets.

3.4. Evaluation Setup

In this work, we have considered a subject-wise cross-validation, which is a version of the k-fold cross-validation procedure where the folds contain recordings from different subjects. In this methodology, the data are divided into k groups or folds to train, validate, and test the system with different data. However, it is assured that all the recordings from the same subject are included only in a fold. In this case, a subset of subjects is used for testing, and data from different subjects are used for training and validation in each iteration. The system is trained, validated, and tested with recordings from different subjects. Once the system model is fitted on the training subset, the validation subset is used for optimizing the model hyperparameters. Finally, the system is evaluated with the testing subset. This process is repeated several times, leaving different subjects for testing in each iteration. The results are the average of the results obtained for all trials (10 folds in this work). This methodology simulates a real-life scenario where the system is evaluated with recordings from subjects different from those used for training.

As evaluation metrics, we have used accuracy, which is defined as the ratio between the number of correctly classified samples and the number of total samples. Considering a classification problem with N testing samples and C classes, accuracy is defined in Equation (2), where P_{ii} refers to the elements of the principal diagonal in the confusion matrix.

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{C} P_{ii} \tag{2}$$

Confidence intervals are used to show statistical significance values and provide confidence about the reliability of the results. These intervals include plausible values for a specific metric. We will assure that there exists a significant difference between the results of two experiments when their confidence intervals do not overlap. Equation (3) represents the computation of confidence intervals attached to a specific accuracy value and N samples when the confidence level is 95%.

$$CI(95\%) = \pm 1.96\sqrt{\frac{accuracy \cdot (100 - accuracy\ )}{N}} \tag{3}$$

4. Results and Discussion

This section describes the datasets used in this study, the experimental setups, the results, and the main discussions.

4.1. Datasets

For this work, we have used six publicly available HAR datasets. The combination of these datasets contains different sensing devices and a wide variety of physical activities, including repetitive movements such as running or walking and postures such as sitting and standing. The datasets used are WISDM_lab (Activity Prediction) [32], WISDM_wild (Actitracker) [33], MotionSense [34], USC-HAD [35,36], PAMAP2 [37], and HARTH [38].

The WISDM_lab dataset contains physical activity from 36 subjects that were carrying a smartphone in their front pants leg pocket. The recording device included an embedded accelerometer sampling at 20 Hz. The data collection was supervised by one of the laboratory team members to ensure the quality of the data. The subjects performed the following activities: walking, jogging, ascending stairs, descending stairs, sitting, and standing. As the samples from ascending and descending stairs were limited, both classes were joined as one: stairs activity. This dataset includes more than 15 h of recorded activity.

The WISDM_wild dataset includes physical activity recordings performed by 209 subjects while wearing a smartphone (HTC Evo model) with an accelerometer sensor using a sampling frequency of 20 Hz. The dataset contains in-the-wild data because the recordings were collected in real conditions without expert supervision. They labeled the data through a drop-down data label chooser in an application. In this context, there were no restrictions about where to wear the device, so the subjects could record the activity while keeping the smartphone inside a shirt pocket or a trousers pocket, even while holding it in the hand. The subjects performed the following activities: walking, jogging, stairs, sitting, standing, and lying down. This dataset includes 40 h of recorded activity.

The MotionSense dataset contains recordings of different physical activities performed by 24 subjects at the Queen Mary University of London's Mile End campus. These participants wore in their trousers' front pocket a smartphone (iPhone 6S model) with an accelerometer sampling at 50 Hz. The subjects performed the following activities: walking downstairs, walking upstairs, sitting, standing, walking, and jogging. This dataset contains 8 h of recordings.

The USC-HAD dataset includes recordings from 14 subjects performing physical activities while wearing an IMU (MotionNode, online: https://sipi.usc.edu/had/mi_ubicomp_sagaware12.pdf, available on 11 June 2023) packed into a pouch and attached to the front right hip. In this case, the sensor orientation variability could be lower thanks to the attachment. This measurement unit included an accelerometer sampling at 100 Hz. The physical activities included in this dataset are walking forward, walking left, walking right, walking upstairs, walking downstairs, running forward, jumping, sitting, standing, sleeping, elevator up, and elevator down. This dataset includes approximately 8 h of recorded activity.

The PAMAP2 dataset contains recordings of different physical activities performed by nine people wearing three IMUs (Inertial Measurement Units) (Trivisio, Yutz, France) with accelerometers sampling at 100 Hz. These units are placed onto three different body locations: chest, wrist on the dominant arm, and ankle on the dominant side. In this case, the sensor orientation variability could be lower thanks to the attachments. The subjects performed the following activities: lying, sitting, standing, walking, running, cycling, Nordic walk, ascending stairs, descending stairs, ironing, vacuum cleaning, and rope jumping. This dataset includes more than 5.5 h of recorded activity.

The HARTH dataset includes recordings of different physical activities performed by 22 people wearing two accelerometer sensors (Axivity AX3 model) sampling at 100 Hz. These units are placed onto the right front thigh (approximately 10 cm above the upper kneecap) and lower back (approximately third lumbar vertebra). This dataset contains data collected under laboratory conditions and in a free-living setting where no further instructions on where and when to carry out the activities but including the sensor attachments. In this case, the sensor orientation variability could be lower thanks to the attachments. The physical activities included in this dataset are walking, running, shuffling (standing with leg movement), ascending stairs, descending stairs, standing, sitting, lying, cycling while sitting, and cycling while standing. The recordings of inactive cycling while sitting (without leg movement) and inactive cycling while standing (without leg movement) were not used in this work because these activities are too unusual. This dataset includes more than 17 h.

Table 1 includes the main characteristics of the datasets used in this work, including the number of subjects, the number of physical activities (repetitive movements and postures), the devices used for recording the motion, their location, and the sampling frequency of the accelerometers.

Table 1. Main characteristics of the datasets used.

Dataset	# Subject	# Activity	# Rep. Mov.	# Posture	Device	Device/Sensor Location	Sampling Rate (Hz)
WISDM_lab	36	5	3	2	Smartphone	Front pants pocket	20
WISDM_wild	209	6	3	3	Smartphone	Free	20
MotionSense	24	6	4	2	Smartphone	Front pants pocket	50
USC-HAD	14	12	7	5	Sensor	Front right hip	100
PAMAP2	9	12	9	3	Sensor	Hand, chest, and ankle	100
HARTH	22	10	7	3	Sensor	Thigh and lower back	100

Table 2 displays the overall duration of recorded physical activity for the datasets used in this work, as well as the time for each specific physical activity within each dataset.

Table 2. Time for each activity within the datasets used characteristics of the datasets used.

Dataset	Total Time (h)	Time per Activity
WISDM_lab	15	Walking (20,970 s), jogging (16,453 s), stairs (11,063 s), sitting (2954 s), and standing (2306 s)
WISDM_wild	40	Walking (60,684 s), jogging (21,813 s), stairs (2515 s), sitting (32,607 s), standing (14,030), and lying down (13,424 s)
MotionSense	8	Walking downstairs (2578 s), walking upstairs (3198 s), sitting (6863 s), standing (6210 s), walking (6987 s), and jogging (2617 s)
USC-HAD	8	Walking forward (3772 s), walking left (2588 s), walking right (2755 s), walking upstairs (2118 s), walking downstairs (1974 s), running forward (1765 s), jumping (1072 s), sitting (2615 s), standing (2360 s), sleeping (3750 s), elevator up (1653 s), and elevator down (1602 s)
PAMAP2	5.5	Lying (1925 s), sitting (1852 s), standing (1899 s), walking (2387 s), running (978 s), cycling (1646 s), Nordic walk (1881 s), ascending stairs (1173 s), descending stairs (1051 s), ironing (1755 s), vacuum cleaning (2387 s), and rope jumping (488 s)
HARTH	17	Walking (11,661 s), running (2917 s), shuffling (standing with leg movement) (1180 s), ascending stairs (817 s), descending stairs (740 s), standing (7327 s), sitting (29,003 s), lying (4285 s), cycling while sitting (3965 s), and cycling while standing (544 s)

4.2. Experimental Setups and Results

Some of our HAR previous works were focused on optimizing the different modules of the HAR system obtaining the highest recognition performance over PAMAP2 [2,3], MotionSense [39], or USC-HAD [1] datasets. This work is focused on showing how the proposed algorithm could correct the recognition errors due to changes in the recording sensor orientation. The hypothesis to demonstrate in these experiments is that the proposed algorithm can compensate the degradations suffered by state-of-the-art HAR systems when random rotations are introduced in the sensors. The goal is to recover the system degradation, obtaining the same performance compared to the baseline experiment (where all the sensors were properly oriented). We considered four different experimental setups or situations to demonstrate this hypothesis:

1. Baseline. First, we used the original data from the datasets for training and testing a state-of-the-art HAR system. Most of the datasets (except for WISDM_wild) were obtained under laboratory conditions; the data collection protocol was controlled by experts and all the recording devices were located using the same orientation; thus, there was no effect due to sensor orientation.
2. Rotated. Second, we included random rotations over the tri-axial accelerometer signals to simulate changes in sensor orientation. These changes were based on the rotation matrix, which performed a transformation in Euclidean space. Since we managed tri-axial signals, we applied the rotation over one out of the three axes that were randomly selected for each subject, keeping the remaining axes fixed. The

rotation matrices used for each axis are included in Equation (4). We performed preliminary studies using different angle values, but no effect was observed, so we finally applied a rotation of θ equal to 45° for this work.

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix};\ R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix};\ R_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

3. Rotated and algorithm. Third, we applied the proposed algorithm to compensate for the random sensor rotations by extracting a consistent reference system and transforming the acceleration from the sensor reference system to the consistent reference system. The same algorithm was applied to all types of activities, including movements and postures.
4. Rotated and algorithm per type of activity. Finally, we repeated the third experimental setup but applying specific approaches depending on the type of activity (movements or postures). We used the approach based on the consistent reference system to movements and the solution of subtracting the gravity for postures.

In cases of developing an HAR system dealing with several types of activities, it would be necessary to include an initial classifier module [2] to distinguish between movements and postures and then apply specific approaches based on the type of activities (computing the new reference system or just subtracting the component of the gravity direction), as shown in Figure 4. Considering this system, an initial classifier was included to automatically detect the type of activity: movements vs. postures. This module was implemented using the same CNN architecture presented in Section 3.3, but with a SoftMax function at the end to classify two classes: movements and postures. This classifier has a very high average accuracy (over 95%) because it deals with a simple classification task. Afterward, each example is processed in a different way depending on this classification. The results presented in this work also include the errors produced by this automatic pre-classification.

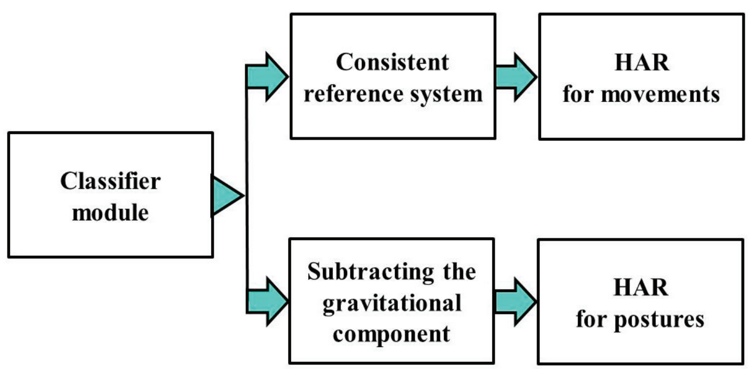


Figure 4. HAR system dealing with different types of activities.

In the case of processing each type of activity separately, we applied the most advantageous approach for each case as mentioned in the algorithm description section. For movements (they have a forward movement direction), we generated a consistent reference system based on the person's movement direction. For postures (as they do not have a motion forward direction), we directly subtracted the gravitational component.

Table 3 includes the accuracy of the experiments in the four experimental setups for all the datasets. Table 3 already includes the impact of the initial classification module to distinguish between movements and postures. We observe that the performance of the results when considering miss-oriented sensors (column "Rotated" of Table 3) decreased significantly compared to the baseline system (column "Baseline" of Table 3); including random rotations over the acceleration signals makes the system performance decrease. For example, in the case of WISDM_lab dataset, we initially reached an accuracy of 91.57 ± 0.23% and decreased to 89.19 ± 0.26% when including random rotations.

Table 3. Results of the four experimental setups for all the datasets. References are included with the descriptions of state-of-the-art HAR systems in each case.

Datasets and State-of-the-Art Systems	Accuracy (%) Depending on the Experimental Setup			
	Baseline	**Rotated (Changes in Sensor Orientation)**	**Rotated and Algorithm**	**Rotated and Different Approaches per Type of Activity**
WISDM_lab [32]	91.57 ± 0.23	89.19 ± 0.26	90.28 ± 0.25	91.36 ± 0.24
WISDM_wild [33]	73.54 ± 0.23	71.17 ± 0.23	71.52 ± 0.23	81.54 ± 0.20
MotionSense [39]	95.48 ± 0.24	88.22 ± 0.37	87.78 ± 0.38	92.09 ± 0.31
USC-HAD [1]	63.56 ± 0.56	58.62 ± 0.58	56.48 ± 0.58	59.35 ± 0.58
PAMAP2—Chest [2,3]	72.29 ± 0.63	65.35 ± 0.67	67.44 ± 0.66	68.12 ± 0.66
PAMAP2—Wrist [2,3]	77.27 ± 0.59	68.52 ± 0.65	70.06 ± 0.64	74.71 ± 0.61
PAMAP2—Ankle [2,3]	70.17 ± 0.64	63.46 ± 0.68	62.50 ± 0.68	67.93 ± 0.66
HARTH—Back [38]	88.58 ± 0.25	81.89 ± 0.30	80.21 ± 0.31	87.52 ± 0.26
HARTH—Thigh [38]	91.67 ± 0.22	83.90 ± 0.29	81.72 ± 0.30	87.56 ± 0.26

In a first attempt, we used the same algorithm for all the activities (column "Rotated and algorithm" of Table 3). We were able to recover part of the degradation, but it was not possible to reach the same accuracy obtained in the baseline (without random rotations). For WISDM_lab dataset, we obtained an accuracy of 90.28 ± 0.25%. Extracting a consistent reference system with postures can be counterproductive because it cannot be extracted properly.

In a second attempt, we decided to apply specific approaches depending on the type of activity: the consistent new reference system for repetitive movements and subtracting the component of the gravity direction for postures. In this case (column "Rotated and different approaches per type of activity" of Table 3), we were able to reach a similar performance to the baseline experimental setup. For example, in the case of WISDM_lab dataset, we obtained an accuracy of 91.83 ± 0.23% that was reduced to 91.36 ± 0.24% when including the initial classifier module (with a 99.90 ± 0.03% of classification accuracy). Adapting the algorithm to the type of activity solved the degradation from sensor orientation. In the case of WISDM_wild, we not only recovered the degradation but also achieved an important improvement in performance compared to the baseline system. One of the reasons for this improvement is that this dataset already contains real data whose collection protocol was not supervised by experts, so the original data were already affected by some rotations that were mitigated by our proposed approach. Only a slight reduction of performance (2.37%) was obtained by comparing the baseline and rotated setup, which means that the original data were noisy and already included sensor rotations.

Table 4 includes the results of the experiments for all the datasets distinguishing types of activity. Results suggest that the errors due to the changes in sensor orientation could be mitigated by applying specific algorithms because the final systems could attain a similar performance compared to the baseline.

Table 4. Results of three experimental setups for the different datasets for each type of activity.

Dataset	Type of Activity	Accuracy (%) Depending on the Experimental Setup		
		Baseline (Supervised by Experts)	Rotated	Rotated and Algorithm per Type of Activity
WISDM_lab	Rep. Mov.	91.41 ± 0.25	88.64 ± 0.28	91.82 ± 0.24
	Postures	96.71 ± 0.48	71.67 ± 1.22	88.17 ± 0.87
WISDM_wild	Rep. Mov.	88.46 ± 0.21	89.18 ± 0.21	91.09 ± 0.19
	Postures	63.76 ± 0.38	47.46 ± 0.40	58.66 ± 0.39
MotionSense	Rep. Mov.	90.98 ± 0.45	87.39 ± 0.52	91.81 ± 0.43
	Postures	98.55 ± 0.21	84.39 ± 0.62	96.57 ± 0.31
USC-HAD	Rep. Mov.	59.72 ± 0.76	55.29 ± 0.77	58.66 ± 0.76
	Postures	67.75 ± 0.84	62.77 ± 0.87	67.19 ± 0.84
PAMAP2—Chest	Rep. Mov.	76.72 ± 0.71	63.38 ± 0.81	73.80 ± 0.74
	Postures	73.93 ± 1.14	57.91 ± 1.28	75.16 ± 1.12
PAMAP2—Wrist	Rep. Mov.	84.42 ± 0.61	73.23 ± 0.74	80.23 ± 0.67
	Postures	71.28 ± 1.18	57.33 ± 1.29	72.27 ± 1.16
PAMAP2—Ankle	Rep. Mov.	80.17 ± 0.67	75.97 ± 0.71	76.07 ± 0.71
	Postures	74.88 ± 1.13	54.88 ± 1.29	64.02 ± 1.25
HARTH—Back	Rep. Mov.	91.71 ± 0.37	88.20 ± 0.43	89.89 ± 0.40
	Postures	88.16 ± 0.31	84.62 ± 0.35	87.85 ± 0.32
HARTH—Thigh	Rep. Mov.	92.79 ± 0.34	87.46 ± 0.44	90.04 ± 0.40
	Postures	93.83 ± 0.23	84.29 ± 0.35	87.48 ± 0.32

Figure 5 displays four confusion matrices for the MotionSense dataset by which to compare different experimental setups. These matrices show the classification results in the rotated setup for the repetitive movements (Figure 5a, related to 87.39 ± 0.52% of accuracy) and postures (Figure 5c, related to 87.39 ± 0.52% of accuracy). Additionally, we show confusion matrices for the rotation and algorithm per type of activity experimental setup: repetitive movements (Figure 5b, related to 91.81 ± 0.43% of accuracy) and postures (Figure 5d, related to 96.57 ± 0.31% of accuracy). It is possible to observe that the confusion for the different classes is reduced when applying our algorithm.

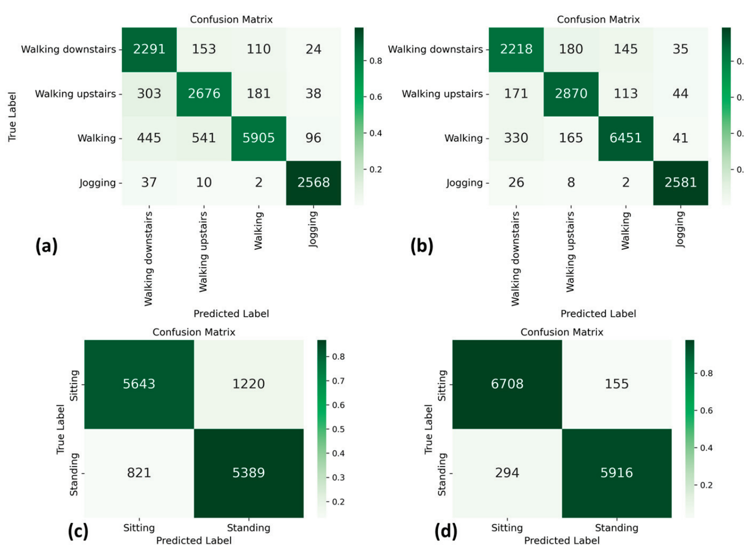


Figure 5. Confusion matrices for repetitive movements and postures classification results in the MotionSense dataset, using rotated experimental setup (**a**,**c**) and rotation and algorithm per type of activity experimental setup (**b**,**d**).

Table 5 details the precision and recall results per activity for these experiments, where it is observed that both metrics of every class, including repetitive movements and postures, increase when applying the algorithm, except for recall of walking downstairs. However, it is important to mention that this class is compared to very similar activities, such as walking upstairs, walking, and jogging.

Table 5. Precision and recall results per activity on two experimental setups for the MotionSense dataset.

Type of Activity	Activity	Results (%) Depending on the Experimental Setup			
		Rotated		Rotated and Algorithm per Type of Activity	
		Precision	Recall	Precision	Recall
Repetitive Movements	Walking downstairs	74.48 ± 1.54	88.87 ± 1.21	80.80 ± 1.47	86.04 ± 1.34
	Walking upstairs	79.17 ± 1.37	83.68 ± 1.28	89.05 ± 1.08	89.74 ± 1.05
	Walking	95.27 ± 0.53	84.51 ± 0.85	96.13 ± 0.46	92.33 ± 0.62
	Jogging	94.20 ± 0.88	98.13 ± 0.52	95.56 ± 0.78	98.62 ± 0.45
Postures	Sitting	87.30 ± 0.81	82.22 ± 0.90	95.80 ± 0.47	97.74 ± 0.35
	Standing	81.54 ± 0.94	86.78 ± 0.84	97.45 ± 0.40	95.27 ± 0.53

4.3. Discussions and Insights

The first insight obtained from results in Table 3 was the important degradation suffered by state-of-the-art HAR systems when random sensor rotations are introduced in the tri-axial accelerations. This important degradation justified the main contribution of this study, namely, the proposal and evaluation of a new pre-processing algorithm to compensate for sensor rotations in state-of-the-art HAR systems.

The proposed algorithm builds a consistent reference system independently of the sensor orientation and represents the acceleration according to this consistent reference system (independently of the sensor orientation). The creation of the consistent reference system is based on the estimation of gravitational and forward movement directions. These orthogonal directions (plus the cross product) form a tri-axial orthogonal system. The acceleration representation is transformed from the sensor axial system to this new reference system, which is more consistent in its movements.

During the experiments, we realized that this algorithm works well for movements such as running or cycling but not for postures such as sitting or standing. In postures, there is no motion in the forward direction, so it is not possible to extract the new reference system. As an alternative, we proposed subtracting the gravity from the total acceleration. This insight was very important because it allowed the combination of several strategies and the design of a complete solution (applicable to all types of activities).

This complete solution has been evaluated with state-of-the-art HAR systems based on deep learning algorithms over six different datasets. These datasets cover a very wide range of activities, subjects, and recording conditions. The experiments section showed the results of four different situations: the baseline situation (where all the sensors were correctly oriented), the rotated scenario (with the introduction of random rotations to the sensors), and two applications of the proposed algorithm (i.e., same strategy for all the activities or differentiating between movements and postures). From the experiments in Table 3, we can conclude that the proposed method is able to recover the degradation produced in the HAR systems when random rotations are introduced in the tri-axial acceleration. This recovery is complete when we apply a different strategy depending on the type of activity (movements or postures). This result is very important because this is the first work proposing a complete solution (for any kind of activity). Another insight obtained from the results is the less degradation in movements compared to postures when including sensor rotations (Table 4): postures are more sensible to changes in sensor orientation.

To compare our system with previous works, we computed the mitigation capability as the percentage of degradation which the algorithm can recover or compensate. The previous work with the best mitigation capability was [19], extracting vertical and horizontal features. The best proposed method was able to recover 86% of the performance decrease due to sensor rotation. In our case, the algorithm proposed in this paper recovered (on average, along six different datasets) 100% of the performance decrement. We observed that the algorithm proposed in this paper was able to deal with severe sensor rotations, especially in locations such as the wrist for the PAMAP2 dataset (see the improvement of 6.19% comparing rotated and last columns in Table 3).

5. Conclusions

Changes in sensor orientation are an important problem that affects the performance of HAR systems in many different applications. In this paper, a new preprocessing algorithm has been proposed to reduce the negative impact of these changes. This algorithm creates a consistent reference system (based on the estimation of gravitational and forward movement directions) and transforms the tri-axial accelerometer signals representation. This algorithm has been very useful for movements; in this case, it is easy to leverage the gravitational and forward component information to create a consistent reference system with which to represent the movement. In the case of postures (sitting or standing), a forward movement vector does not exist, and it cannot be used for extracting the consistent reference system. In these cases, subtracting the gravitational component of the signals has been more useful.

The proposed approach was included in a preprocessing module (i.e., before the signal processing module) of a state-of-the-art HAR system and evaluated over six different HAR datasets that include repetitive movements and postures. We used a subject-wise cross-validation methodology: different subjects were used for training, validation, and testing the system in each iteration. For the WISDM dataset, the sensor orientation errors reduced the classification accuracy from $91.56 \pm 0.23\%$ to $89.19 \pm 0.26\%$. This performance decrease was mitigated with the proposed algorithm, increasing the accuracy to $91.46 \pm 0.30\%$ when applying specific approaches depending on the type of activity, and reaching the same results that those achieved with the sensor correctly oriented.

However, this study has a limitation: the current algorithm is applied over isolated sensors, so it could be interesting to deal with several sensors at the same time. An interesting solution could be to estimate the consistent reference system from one sensor and then use this system for all the sensors. Another interesting future work could be the analysis of the best sensor location to estimate the consistent reference system. Finally, we would like to apply the proposed algorithm to data with real device orientation changes.

Author Contributions: Conceptualization, M.G.-M. and R.S.-S.; methodology, M.G.-M., F.F.-M. and R.S.-S.; software, M.G.-M. and J.L.-I.; validation, M.G.-M., F.F.-M. and R.S.-S.; formal analysis, M.G.-M. and J.L.-I.; investigation, M.G.-M. and J.L.-I.; resources, R.S.-S.; data curation, J.L.-I.; writing—original draft preparation, M.G.-M.; writing—review and editing, M.G.-M., J.L.-I., F.F.-M. and R.S.-S.; visualization, M.G.-M.; supervision, F.F.-M. and R.S.-S.; project administration, F.F.-M. and R.S.-S.; funding acquisition, F.F.-M. and R.S.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in different repositories: WISDM_lab (Activity Prediction) (https://www.cis.fordham.edu/wisdm/dataset.php), WISDM_wild (Actitracker) (https://www.cis.fordham.edu/wisdm/dataset.php), MotionSense (https://www.kaggle.com/datasets/malekzadeh/motionsense-dataset), USC-HAD (https://sipi.usc.edu/had/) [36], PAMAP2 (10.24432/C5NW2H), and HARTH (10.24432/C5NC90).

Acknowledgments: The work leading to these results was supported by the Spanish Ministry of Science and Innovation through the project AMIC-PoC, BeWord, GOMINOLA, and TremorDetect: Detección de la enfermedad de Parkinson a través de señales inerciales (PDC2021-120846-C42, PID2021-126061OB-C43, PID2020-118112RB-C21, and PID2020-118112RB-C22 funded by MCIN/AEI/10.13039/501100011033, by E.T.S.I. de Telecomunicación, and by the European Union "NextGenerationEU/PRTR")'. We gratefully acknowledge the support of the NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gil-Martin, M.; San-Segundo, R.; Fernandez-Martinez, F.; Ferreiros-Lopez, J. Time Analysis in Human Activity Recognition. *Neural Process. Lett.* **2021**, *53*, 4507–4525. [CrossRef]
2. Gil-Martín, M.; San-Segundo, R.; Fernández-Martínez, F.; de Córdoba, R. Human activity recognition adapted to the type of movement. *Comput. Electr. Eng.* **2020**, *88*, 106822. [CrossRef]
3. Gil-Martin, M.; San-Segundo, R.; Fernandez-Martinez, F.; Ferreiros-Lopez, J. Improving physical activity recognition using a new deep learning architecture and post-processing techniques. *Eng. Appl. Artif. Intell.* **2020**, *92*, 103679. [CrossRef]
4. Pires, I.M.; Hussain, F.; Marques, G.; Garcia, N.M. Comparison of machine learning techniques for the identification of human activities from inertial sensors available in a mobile device after the application of data imputation techniques. *Comput. Biol. Med.* **2021**, *135*, 104638. [CrossRef]
5. Islam, M.M.; Nooruddin, S.; Karray, F.; Muhammad, G. Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects. *Comput. Biol. Med.* **2022**, *149*, 106060. [CrossRef]
6. Qiu, S.; Zhao, H.K.; Jiang, N.; Wang, Z.L.; Liu, L.; An, Y.; Zhao, H.Y.; Miao, X.; Liu, R.C.; Fortino, G. Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges. *Inf. Fusion* **2022**, *80*, 241–265. [CrossRef]
7. Munoz-Organero, M. Outlier Detection in Wearable Sensor Data for Human Activity Recognition (HAR) Based on DRNNs. *IEEE Access* **2019**, *7*, 74422–74436. [CrossRef]
8. Morshed, M.G.; Sultana, T.; Alam, A.; Lee, Y.K. Human Action Recognition: A Taxonomy-Based Survey, Updates, and Opportunities. *Sensors* **2023**, *23*, 2182. [CrossRef]
9. Hsu, Y.; Yang, S.; Chang, H.; Lai, H. Human Daily and Sport Activity Recognition Using a Wearable Inertial Sensor Network. *IEEE Access* **2018**, *6*, 31715–31728. [CrossRef]
10. Zhuang, Z.; Xue, Y. Sport-Related Human Activity Detection and Recognition Using a Smartwatch. *Sensors* **2019**, *19*, 5001. [CrossRef]
11. Hendry, D.; Chai, K.; Campbell, A.; Hopper, L.; O'Sullivan, P.; Straker, L. Development of a Human Activity Recognition System for Ballet Tasks. *Sport. Med.-Open* **2020**, *6*, 10. [CrossRef] [PubMed]
12. Gil-Martin, M.; Johnston, W.; San-Segundo, R.; Caulfield, B. Scoring Performance on the Y-Balance Test Using a Deep Learning Approach. *Sensors* **2021**, *21*, 7110. [CrossRef] [PubMed]
13. Sardar, A.W.; Ullah, F.; Bacha, J.; Khan, J.; Ali, F.; Lee, S. Mobile sensors based platform of Human Physical Activities Recognition for COVID-19 spread minimization? *Comput. Biol. Med.* **2022**, *146*, 105662. [CrossRef]
14. Slim, S.O.; Atia, A.; Elfattah, M.M.A.; Mostafa, M.-S.M. Survey on Human Activity Recognition based on Acceleration Data. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 84–98. [CrossRef]
15. Elbasiony, R.; Gomaa, W. A Survey on Human Activity Recognition Based on Temporal Signals of Portable Inertial Sensors. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 921, pp. 734–745.
16. Tan, T.-H.; Wu, J.-Y.; Liu, S.-H.; Gochoo, M. Human Activity Recognition Using an Ensemble Learning Algorithm with Smartphone Sensor Data. *Electronics* **2022**, *11*, 322. [CrossRef]
17. Ma, L.; Huang, M.; Yang, S.; Wang, R.; Wang, X. An Adaptive Localized Decision Variable Analysis Approach to Large-Scale Multiobjective and Many-Objective Optimization. *IEEE Trans. Cybern.* **2022**, *52*, 6684–6696. [CrossRef]
18. Janidarmian, M.; Fekr, A.R.; Radecka, K.; Zilic, Z. A Comprehensive Analysis on Wearable Acceleration Sensors in Human Activity Recognition. *Sensors* **2017**, *17*, 529. [CrossRef]
19. Yang, J. Toward physical activity diary: Motion recognition using simple acceleration features with mobile phones. In Proceedings of the 1st International Workshop on Interactive Multimedia for Consumer Electronics, Beijing, China, 23 October 2009; pp. 1–10.
20. San-Segundo, R.; Blunck, H.; Moreno-Pimentel, J.; Stisen, A.; Gil-Martn, M. Robust Human Activity Recognition using smartwatches and smartphones. *Eng. Appl. Artif. Intell.* **2018**, *72*, 190–202. [CrossRef]
21. Henpraserttae, A.; Thiemjarus, S.; Marukatat, S. Accurate Activity Recognition Using a Mobile Phone Regardless of Device Orientation and Location. In Proceedings of the 2011 International Conference on Body Sensor Networks, Dallas, TX, USA, 23–25 May 2011; pp. 41–46.
22. Yurtman, A.; Barshan, B. Activity Recognition Invariant to Sensor Orientation with Wearable Motion Sensors. *Sensors* **2017**, *17*, 1838. [CrossRef]

23. Hernandez Sanchez, S.; Fernandez Pozo, R.; Hernandez Gomez, L.A. Estimating Vehicle Movement Direction from Smartphone Accelerometers Using Deep Neural Networks. *Sensors* **2018**, *18*, 2624. [CrossRef]
24. Xia, X.; Hashemi, E.; Xiong, L.; Khajepour, A. Autonomous Vehicle Kinematics and Dynamics Synthesis for Sideslip Angle Estimation Based on Consensus Kalman Filter. *IEEE Trans. Control Syst. Technol.* **2023**, *31*, 179–192. [CrossRef]
25. Liu, W.; Xia, X.; Xiong, L.; Lu, Y.S.; Gao, L.T.; Yu, Z.P. Automated Vehicle Sideslip Angle Estimation Considering Signal Measurement Characteristic. *IEEE Sens. J.* **2021**, *21*, 21675–21687. [CrossRef]
26. Gao, L.T.; Xiong, L.; Xia, X.; Lu, Y.S.; Yu, Z.P.; Khajepour, A. Improved Vehicle Localization Using On-Board Sensors and Vehicle Lateral Velocity. *IEEE Sens. J.* **2022**, *22*, 6818–6831. [CrossRef]
27. Liu, W.; Quijano, K.; Crawford, M.M. YOLOv5-Tassel: Detecting Tassels in RGB UAV Imagery With Improved YOLOv5 Based on Transfer Learning. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8085–8094. [CrossRef]
28. Mizell, D. Using gravity to estimate accelerometer orientation. In Proceedings of the Seventh IEEE International Symposium on Wearable Computers, White Plains, NY, USA, 21–23 October 2003; pp. 252–253. [CrossRef]
29. Gil-Martin, M.; San-Segundo, R.; Fernando D'Haro, L.; Manuel Montero, J. Robust Motion Biomarker for Alcohol Consumption. *IEEE Instrum. Meas. Mag.* **2022**, *25*, 83–87. [CrossRef]
30. Tang, C.I.; Perez-Pozuelo, I.; Spathis, D.; Brage, S.; Wareham, N.J.; Mascolo, C. SelfHAR: Improving Human Activity Recognition through Self-training with Unlabeled Data. *arXiv* **2021**, arXiv:2102.06073. [CrossRef]
31. San-Segundo, R.; Manuel Montero, J.; Barra-Chicote, R.; Fernandez, F.; Manuel Pardo, J. Feature extraction from smartphone inertial signals for human activity segmentation. *Signal Process.* **2016**, *120*, 359–372. [CrossRef]
32. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl.* **2011**, *12*, 74–82. [CrossRef]
33. Lockhart, J.W.; Weiss, G.M.; Xue, J.C.; Gallagher, S.T.; Grosner, A.B.; Pulickal, T.T. Design considerations for the WISDM smart phone-based sensor mining architecture. In Proceedings of the Fifth International Workshop on Knowledge Discovery from Sensor Data, San Diego, CA, USA, 21 August 2011; pp. 25–33.
34. Malekzadeh, M.; Clegg, R.G.; Cavallaro, A.; Haddadi, H. Protecting Sensory Data against Sensitive Inferences. In Proceedings of the 1st Workshop on Privacy by Design in Distributed Systems, Porto, Portugal, 23–26 April 2018. [CrossRef]
35. Zhang, M.; Sawchuk, A.A.; Assoc Comp, M. USC-HAD: A Daily Activity Dataset for Ubiquitous Activity Recognition Using Wearable Sensors. In Proceedings of the Ubicomp'12: Proceedings of the 2012 Acm International Conference on Ubiquitous Computing, Pittsburgh, PA, USA, 5–8 September 2012; pp. 1036–1043.
36. Zhang, M. The USC-SIPI Human Activity Dataset. Available online: http://sipi.usc.edu/had/ (accessed on 1 March 2022).
37. Reiss, A.; Stricker, D. Creating and benchmarking a new dataset for physical activity monitoring. In Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments, Heraklion, Crete, Greece, 29 June–1 July 2022; p. 40.
38. Logacjov, A.; Bach, K.; Kongsvold, A.; Bårdstu, H.B.; Mork, P.J. HARTH: A Human Activity Recognition Dataset for Machine Learning. *Sensors* **2021**, *21*, 7853. [CrossRef]
39. Gil-Martín, M.; Sánchez-Hernández, M.; San-Segundo, R. Human Activity Recognition Based on Deep Learning Techniques. *Proceedings* **2020**, *42*, 15.

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Sensors Editorial Office
E-mail: sensors@mdpi.com
www.mdpi.com/journal/sensors

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-3676-5