Hindawi
Journal of Sensors
Volume 2022, Article ID 9930681, 14 pages
https://doi.org/10.1155/2022/9930681

Research Article

Bluetooth-Low-Energy-Based Fall Detection and Warning System for Elderly People in Nursing Homes

Nick De Raeve, Adnan Shahid, Matthias de Schepper, Eli De Poorter, Ingrid Moerman, Jo Verhaevert, Patrick Van Torre, and Hendrik Rogier

IDLab, Department of Information Technology, Ghent University-imec, Technologiepark-Zwijnaarde 126, 9052 Ghent, Belgium

Correspondence should be addressed to Nick De Raeve; nick.deraeve@ugent.be

Received 11 March 2021; Revised 30 November 2021; Accepted 14 December 2021; Published 17 January 2022

Academic Editor: Everardo Vargas-Rodriguez

Due to the ever growing population of elderly people, there is a dramatic increase in fall accidents. Currently, multiple ideas exist to prevent the elderly from falling, by means of technology or individualised fall prevention training programs. Most of them are costly, difficult to implement or less used by the elderly, and they do not deliver the required results. Furthermore, the increasingly older population will also impact the workload of the medical and nursing personnel. Therefore, we propose a novel fall detection and warning system for nursing homes, relying on Bluetooth Low Energy wireless communication. This paper describes the hardware design of a fall-acceleration sensing wearable for the elderly. Moreover, the paper also focuses on a novel algorithm for real-time filtering of the measurement data as well as on a strategy to confirm the detected fall events, based on changes in the person's orientation. In addition, we compare the performance of the algorithm to a machine learning procedure using a convolutional neural network. Finally, the proposed filtering technique is validated via measurements and simulation. The results show that the proposed algorithm as well as the convolutional neural network both results in an excellent accuracy when validating on a common database.

1. Introduction

Fall accidents are a large risk in the life of elderly people and form one of the most important public health problems in the ageing population [1]. Research shows that, from the age of 65 on, the number of fall accidents rises dramatically [2, 3], often with a lethal ending. Furthermore, it is found that individualised fall prevention training programs [4, 5] do not have the intended results. All these fall injuries have a large influence on the health system [6]. Moreover, they definitely impact the workload of the medical and nursing personnel, since they have to constantly check if an elderly person has fallen. This causes stress for the personnel and can lead to mental or physical problems. The current literature proposes multiple solutions to lower the number of fall accidents with lethal endings, by alerting the personnel faster. Accordingly, the workload of the medical and nursing personnel is lowered as well. However, many of these solutions are very expensive, difficult to implement, or hindered by privacy regulations. Therefore, we propose a low-cost fall detection and warning system for nursing homes based on Bluetooth Low Energy 5.1 (BLE 5.1) [7]. The proposed system consists of four different nodes, all using BLE 5.1 or Bluetooth Mesh as a wireless communication platform.

The novel contributions of this paper are as follows. A truly wearable node for fall detection is developed, complemented with victim localization and staff alerting functionality and based on a single-chip solution, exploiting BLE 5.1. Thereby, a novel reliable physics-based fall detection algorithm that does not require any training is conceived, and the proposed algorithm is compared with machine learning (ML) using convolutional neural networks (CNN).

This paper describes the design and implementation of a fall detection wearable for the elderly. First, an overview is presented of the current literature, followed by a detailed explanation of the hardware implementation and algorithm used to reliably detect and confirm a fall. The detection is triggered by acceleration, and the conformation is based on

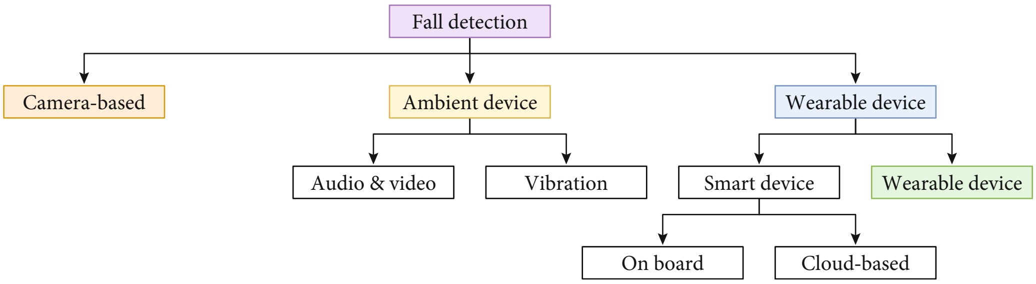

Figure 1: Classification of fall detection systems.

orientation change. Next, there is an overview of all measurements and simulations that are used to construct and validate the algorithm, followed by measurement analysis and conclusions.

2. Related Work

In the current literature, multiple fall detection systems are proposed [8–14]. These detection systems can be split up in three different groups, as can be seen in Figure 1.

A first group is formed by camera-based systems [15–19]. By means of a camera implemented in a room or on a person, a fall is detected based on an extensive algorithm that runs on a PC. Even though these systems work well, they have some disadvantages. Most of them have a limited coverage area, have a very high cost, and can be hindered by privacy regulations. Furthermore, it is important to notice that none of these systems communicate to provide the necessary help when a fall occurs. Since these systems rely on a different principle, we do not go into further detail.

A second group consists of ambient device systems [8–10, 20–23]. This group can be split up into two subgroups. The first subgroup utilises vibration to determine whether a fall occurred. Sensors are mounted in a room, on a floor [21], etc., to monitor the environment and to detect a fall. Accelerometers are integrated into the walls [20] and continuously scanned for vibration, while an algorithm detects a fall based on these wall sensors. The second subgroup attempts to fuse audio and visual data. The system proposed in [23] combines accelerometer and camera data to determine whether a fall has occurred. Just like camera-based systems, these have an even higher implementation cost. Moreover, they can only detect a fall, but they cannot determine which elderly person has fallen, nor can they send vital parameters to the nursing personnel. Most of these systems are prone to generating false alerts, which is stressful for the personnel and can lead to carers paying less attention to the alerts.

A final group of solutions consists of wearable systems. We have opted to divide these systems in two subgroups. The first and biggest subgroup consists of systems that are based on a smart device with integrated accelerometer or gyroscope sensors [24–29]. In a next step, these systems can be divided into categories, based on where most of the processing is performed. In the systems proposed in [24, 26], the accelerometer data are transferred to the cloud or to a PC, where a postprocessing algorithm detects a fall. In the cloud, multiple algorithm types can be used to determine whether a person has fallen. In the current literature, a lot of research is performed on machine learning algorithms [30–39] that detect falls. However, these systems require a lot of training data, which is difficult to obtain. Furthermore, such systems consume a lot of power. Secondly, there are the smart device systems that perform all necessary calculations on the device itself [27, 28]. In this way, there is no unnecessary transfer of data to the cloud.

The second type of wearable device is the simplest in its form. Note that research confirms that elderly people prefer not to wear (obtrusive) smart devices [40, 41]. A truly unobtrusive wearable [42–46] generally only contains a microcontroller (MCU) and an accelerometer or gyroscope sensor. The MCU will determine whether a fall has occurred, based on movement or posture changes of the person wearing the device. The device is energy-efficient and consumes little energy, and the wearable does not hinder the elderly people. Since these wearables do not look like smart devices or do not require any input actions from elderly people, the likelihood of acceptance increases. However, such wearables often do not communicate with the carers. A selected group of fall detection systems uses a gyroscope [47–52] to detect a fall or even the typical movements just before a fall occurs.

There are also several commercially available products. Most of these products [53–55] will make a call to the emergency services when a fall event is observed. In most situations, the detection is based on accelerometer or gyroscope data that are fused together. The detection occurs with a smartphone, smartwatch, or dedicated wearable. Disadvantages of these systems are the smart technology. It is a known fact [40, 41] that elderly do not like to wear smart devices, especially when they are not unobtrusive. There are also some radar-based solutions [56, 57]. These are placed inside the room and detect falling persons. When a fall occurs, the system calls the emergency service. Radar solutions have promising effects but can suffer image clutter caused by furniture in the room. Other products do not only detect when a person falls but also try to prevent a serious fall. The product proposed in [58] is a belt worn around the waist that uses a fusion of different sensors to detect if a person is going to fall. If so, an airbag is deployed and a predefined phone number is called. Compared to our solution, it is very difficult to wear this wearable in an unobtrusive way.

Since the system proposed in this paper only has to detect whether a fall occurred, we opted to use only an accelerometer instead of a gyroscope. The paper will further explain how the accelerometer detects the fall acceleration as well as changes in orientation of the falling person.

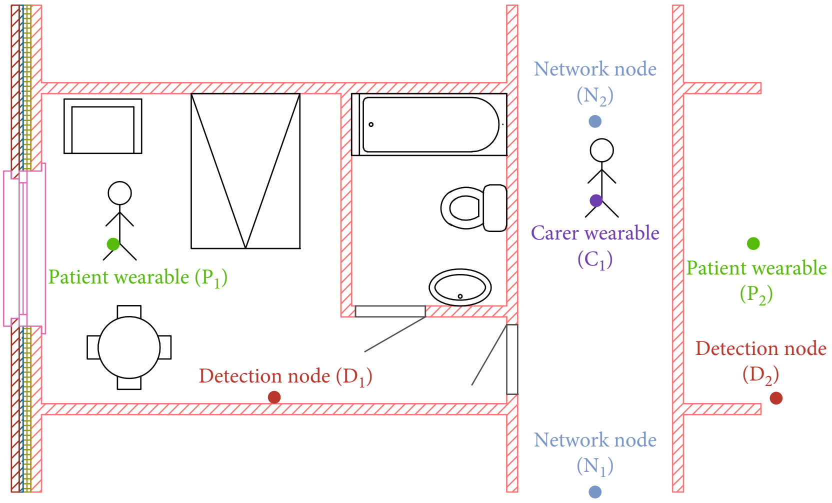

Figure 2: Ground plan of a room in a nursing home with the different nodes mounted. The patient wearable (green dot) is worn by an elderly person who has fallen. The detection node (red dot) is mounted in a central position and communicates with the network nodes (blue dot) mounted in the hallway. These nodes transfer messages to the carers equipped with a wearable/smartphone (magenta dot).

3. Design

Based on the advantages and disadvantages of the previously mentioned systems, the following system requirements for the patient wearable were adopted:

(1) More than 90% of the falls detected

(2) Less than 10% false alerts

(3) Compact and unobtrusive

(4) Battery lifetime of up to 7 days

The proposed system consists of four different nodes, Figure 2 provides a ground plan of a typical room in a nursing home that is equipped with the system, and Figure 3 illustrates the communication steps between the different nodes. The first node, the patient wearable (green dot), is worn around the waist. This node uses BLE 5.1 as a wireless communication platform and contains an accelerometer for measuring the movements of the elderly person. As soon as the node measures a fall, it starts advertising alert messages. For each alert, five subsequent advertisement packets are transmitted to ensure reliable reception. With this approach, system pairing and self-healing are not necessary.

The second node is the detection node (red dot) and is placed in a central position of the room and scans for alert messages from the wearable. When detecting an alert, it pushes a message on the mesh network. This network is realised by the network nodes (blue dots) that are placed in the corridors. These nodes transfer the alert messages to the closest carer wearable (magenta dot), alerting the carer that patient X has fallen in room Y. Since each room is equipped with its detection node, it is easy to map each node to a floor and room number. Additional information can be added to these alert messages or a continuous stream can be setup to visualise the vital parameters from the elderly person. Furthermore, by implementing the proxy feature [59, 60] in the mesh network, every BLE device can communicate on the network. In this way, the carer's wearable can be replaced by a smartphone application.

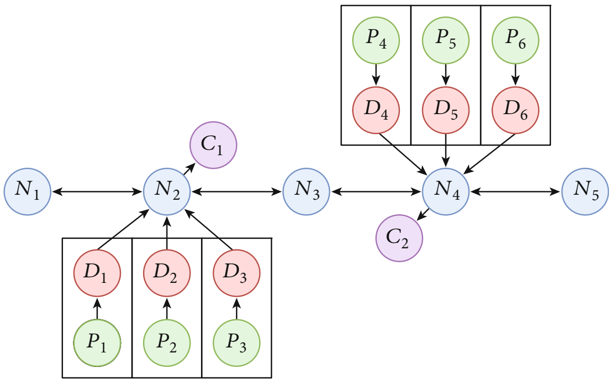

Figure 3: Graphical representation of the communication steps. The patient (P) wearable, represented in green, will detect a fall and send a message to the detection (D) node, represented in red. The number of the detection node corresponds to the room number. This node transmits an alert message to the network (N) nodes, shown in blue. These nodes forward the message to the closest carer (C) node, shown in magenta. The carer can then perform the necessary actions.

The proposed system is very energy efficient, allowing the patient wearable and carer node to be powered by batteries, it does not cause any issues with privacy, the installation costs are minimal, and it is much cheaper than camera systems or extensive individualised training programs. Additionally, there is no need for advanced positioning systems or cellular communications, which would make such a system more expensive. Furthermore, since this network allows connection to smartphones, the nurses do not have to carry extra devices. The following subsections describe the hardware design and the detection algorithm that is executed in the wearable, worn by the elderly person.

3.1. Hardware Design. A number of specific requirements are imposed during the design process of the wearable. First, the wearable has to be unobtrusive to the end-user. Studies show that elderly do not easily adopt novel technology, especially when devices are too big or too complicated to use [40, 41]. Furthermore, based on research performed in [61, 62], the ideal position to mount this wearable is on the waist, which makes the unobtrusiveness even more important. Furthermore, the wearable needs to be energy efficient. The final requirement is the versatility. If the wearable is easy to configure with different sensors, it is more useful for doctors and carers to track or monitor patients in critical conditions. As mentioned before, BLE5.1 is an excellent choice as a wireless communication platform, given its high energy efficiency [63]. Next, the use of the Generic Attribute (GATT) profiles [64], which are part of the BLE stack, makes it very easy to add more functionality. Furthermore, BLE 5.1 has a high message capacity and can operate over a larger range compared to previous versions of BLE.

Figure 4 displays the layout of the printed circuit board (PCB) of the designed wearable. The design relies on the BGM13P wireless module from Silicon Laboratories [65]. This device uses the BLE 5.1 stack and contains an onboard 32-bit 38.4 ARM [66] Cortex-M4 MCU with DSP instruction set. The ARM Cortex-M4 is a modern general-purpose microcontroller employed in many low-power systems. 32-bit processing enables efficient execution of the complex procedures necessary to deploy current-day wireless communication standards such as BLE. The ARM can handle this in a very power efficient way, compared to low-end 8-bit processors, which need much more instructions to achieve the same result. Moreover, modern ARM controllers also provide many power saving modes. Hence, in this application, the processor goes into sleep mode for a relatively long time between short bursts of activity, resulting in an energy efficient solution.

The BGM13P unit also includes an integrated antenna. The complete module has an overall size of 15 × 13 mm, which makes the wearable unobtrusive and much smaller than most smart devices, as is shown in Figure 5. To achieve an optimal communication range, the data sheet of the BGMP13P prescribes an empty space of at least 17 mm near the antenna. To ensure that the device remains unobtrusive to the wearer, the width of the PCB at both lateral sides of the SoC was reduced to 9.5 mm, yielding a decrease by 10% in maximum communication range, according to the data sheet. Furthermore, a TAG-connect [67] connector was used to program all different nodes via the Serial Wire Debug (SWD) protocol [68]. The power supply is a standard small battery of 3.3 V; a power analysis follows in Section 5. Based on previous experience, these modules have been selected to implement this proof of concept.

To measure the movements of the patient, the ultra-small, low-power, triaxial Bosch BMA280 [69] accelerometer was added. This sensor features an integrated low-pass filtering with a filtered-output data rate up to 500 Hz and an unfiltered data rate of 2 kHz. Furthermore, it possesses a resolution of 4096 LSB/g in the ±2g range, can be used in ±2g, ±4g, ±8g, and ±16g range, communicates via an SPI 3-wire communication link, and contains an integrated temperature sensor. These properties make the accelerometer excellent to detect when a patient has fallen. The integrated temperature sensor is a valuable extra feature because if an elderly person falls and is lying on a cold floor, the body temperature can drop to a critical value and lead to hypothermia. By wearing this wearable under clothing, close to the body, the body temperature significantly influences the on-board temperature sensor and a decreasing body temperature can be detected. Taking all these measures into account, the patient's wearable is compact and unobtrusive. Hence, the third design requirement is met.

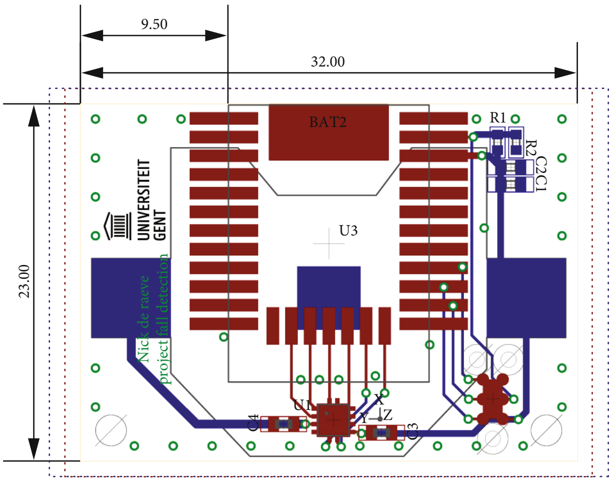

Figure 4: Designed PCB (dimensions in mm).

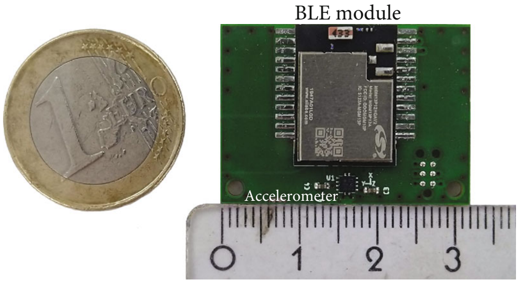

Figure 5: Fabricated PCB with a size of 32 × 23 × 10.

3.2. Algorithm Design and Filtering Procedure. Since the node is constantly receiving the X, Y, and Z acceleration values from the accelerometer at a high sample rate, filtering is necessary to reduce the amount of data. Figure 6 illustrates the flowchart of the applied filtering and decision tree. The algorithm can be split up in two branches. The first branch in red decides whether a fall occurred, based on the amplitude of the accelerometer data, and is called the acceleration branch. The second branch in green takes a decision based on the change of the spatial orientation and is called the

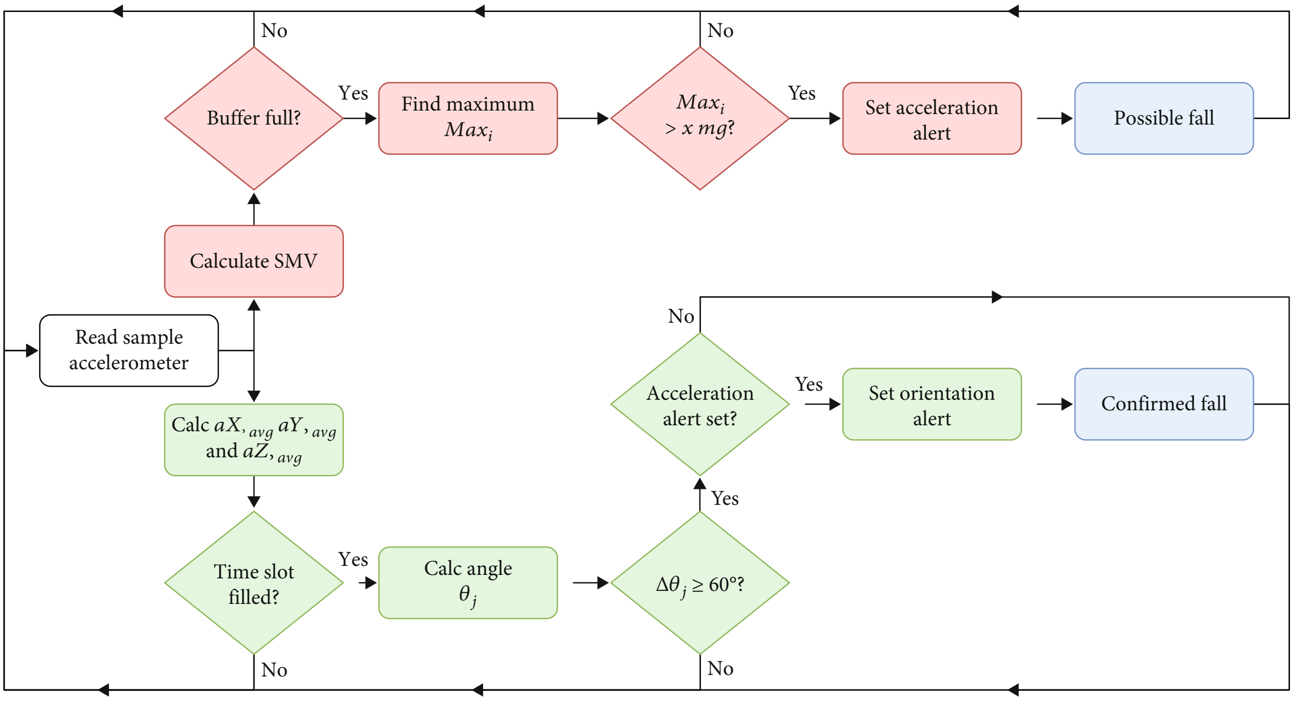

Figure 6: Flowchart of two branches used to filter accelerometer data and determine fall alerts. Acceleration branch coloured in red, orientation branch in green, and decision tree in blue. Note that orientation change alone can never trigger an alert.

orientation branch. At the end of each branch, in the decision tree, a measured fall is classified as a possible or confirmed fall, visualised in blue.

The analysis starts when the samples of the accelerometer are received. The acceleration branch calculates the Signal Magnitude Vector (SMV) as used in many related publications [8, 26–29, 32, 70] based on formula (1) and is expressed in mg = [per mode = symbol]9, 81$e - 3$, where a_X, a_Y, and a_Z are the measured accelerations expressed in mg along the X, Y, and Z axis, respectively. The calculated SMV is stored in a buffer. When this buffer is full, the maximum SMV value is selected. This value has to be larger than a certain threshold value, which will be determined in Section 4.2. When this conditions is valid, the acceleration alert flag is set.

$$\mathrm{SMV} = \sqrt{a_X^2 + a_Y^2 + a_Z^2}. \tag{1}$$

The orientation branch calculates the average a_X, a_Y, and a_Z based on a single exponential smoothing algorithm, as is expressed in formula (2) [71–73], with $a_{\text{axis,avg},i}$ the average calculated for sample i along a certain axis expressed in mg, with $a_{\text{axis,avg},i-1}$ the previously calculated average along that axis, with $a_{\text{axis},i}$ a new sample along the same axis, and with α a damping factor, which is set to 0.1. Based on formulas (3) and (4), for the given sample rate (Section 4.2), this α value results in a time constant τ of 47.45 ms, which leads to a cut-off frequency of 3.35 Hz. Hence, (2) implements a low-pass filtering operation to reduce the noise of the accelerometer samples.

$$a_{\text{axis,avg},i} = (1-\alpha).a_{\text{axis,avg},i-1} + \alpha.a_{\text{axis},i}, \tag{2}$$

$$\alpha = 1 - e^{\frac{-\Delta T}{\tau}} \Rightarrow \tau = \frac{-\Delta T}{\ln(1-\alpha)} = 47,45\,\text{ms}, \tag{3}$$

$$f_c = \frac{1}{2.\pi.\tau} = 3,35\,\text{Hz}. \tag{4}$$

For every 250 ms of captured data, the calculated averages $a_{X,\text{avg}}$, $a_{Y,\text{avg}}$, and $a_{Z,\text{avg}}$ determine a vector $\vec{v}_j$ as in equation (5). Next, based on equation (6) the angle $\Delta\theta_j$ between of the new vector $\vec{v}_j$ and the previous vector $\vec{v}_{j-1}$ is calculated. When this $\Delta\theta_j$ is larger than or equal to 60°, the orientation alert flag is set, but only if the acceleration flag was set earlier. In a final step, the alert flags are checked. When both flags are set, the person definitely fell. Hence, a confirmed fall alert is sent to the carer's wearable. When only one of the flags is set, a normal fall alert is sent. Note that the latter case should still be interpreted as a very high probability of a fall event, although the fall is potentially less severe.

$$\vec{v}_j = a_{X,\text{avg}} \cdot \vec{X} + a_{Y,\text{avg}} \cdot \vec{Y} + a_{Z,\text{avg}} \cdot \vec{Z}, \tag{5}$$

$$\Delta\theta_j = \arccos \frac{\vec{v}_{j-1} \cdot \vec{v}_j}{\left\|\vec{v}_{j-1}\right\| \cdot \left\|\vec{v}_j\right\|}. \tag{6}$$

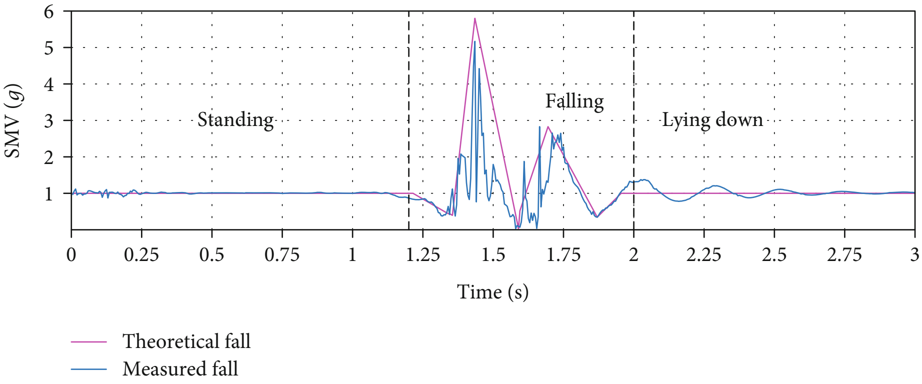

Figure 7: Graphical representation of the calculated SMV for a measured fall at a sample interval of 5 ms compared to a fitted theoretical fall [70]. Both falls represent acceleration data of a test person standing before a fall, during the fall, and lying down after a fall occurred.

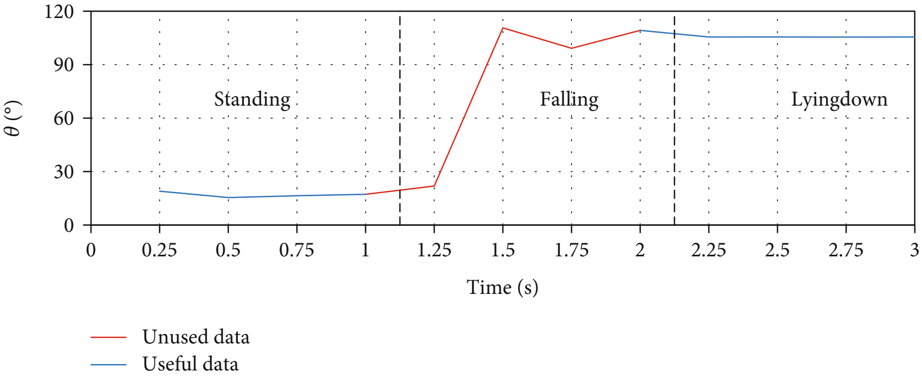

Figure 8: Moving average of the calculated θ in a window size of 250 ms. θ is the angle between the spatial orientation vector of a test person and the gravitational vector, sampled at an interval of 5 ms. This figure illustrates that θ is fluctuating during a fall, but has a stable value before and after a fall occurred.

The main idea behind these branches and the associated conditions originates from looking at the amplitude flow of a fall (Figure 7) and the change in spatial orientation (Figure 8). Figure 7 illustrates the calculated SMV of a measured fall, with a theoretical fall profile fitted on the measurement data. The theoretical fall profile was described in [70]. Under normal conditions, the SMV is around 1 g or 9.81 m/s^2. When a fall occurs, the SMV first drops, followed by a steep positive peak, followed by a smaller peak, and ending near 1 g again. Nearly always, the fall results in a different orientation, as can be seen in Figure 8.

The purpose of this wearable is to provide reliable fall detection without false positives. This is achieved by taking into account that the new maximum value should be larger than the previous maximum and that the difference between the two values should be larger than a predefined threshold expressed in mg. Employing this method, we are able to reliably find the peak and set the acceleration alert flag. Research performed in [70] illustrates that while walking, ascending, and descending stairs, there will never be a peak as large as when a fall occurs.

The main idea of the orientation branch is to confirm a fall by determining the different spatial orientations of the person before and after the fall event. The peak of the angle between two vectors measured before and after the fall with a time interval of 1 in between is used to determine this condition. It is important that this time separation is large enough in order not to include accelerations measured during the fall, as this will disturb the orientation measurement. During the orientation measurement, the total acceleration should be predominantly caused by gravity, which can be confirmed by an SMV value approaching 1 g.

By calculating the moving averages of a_X, a_Y, and a_Z with a window size of 250 and transforming it to a vector, a more reliable orientation angle is obtained. This angle approaches the orientation of the gravity vector and is considered a measurement of the gravity vector. The angle between two measured gravity vectors, taken 250 before the fall and 250 after the fall, provides valuable information to confirm the fall event.

An orientation difference around 90 is expected when a person's orientation changes from standing to lying down. However, a lower threshold value of 60 is proposed in order to account for situations where the elderly person falls on a nearby object and is not lying entirely flat. Note that the orientation alert flag is only set if an acceleration alert flag had already been set. Hence, an orientation change alone can never trigger a fall alert.

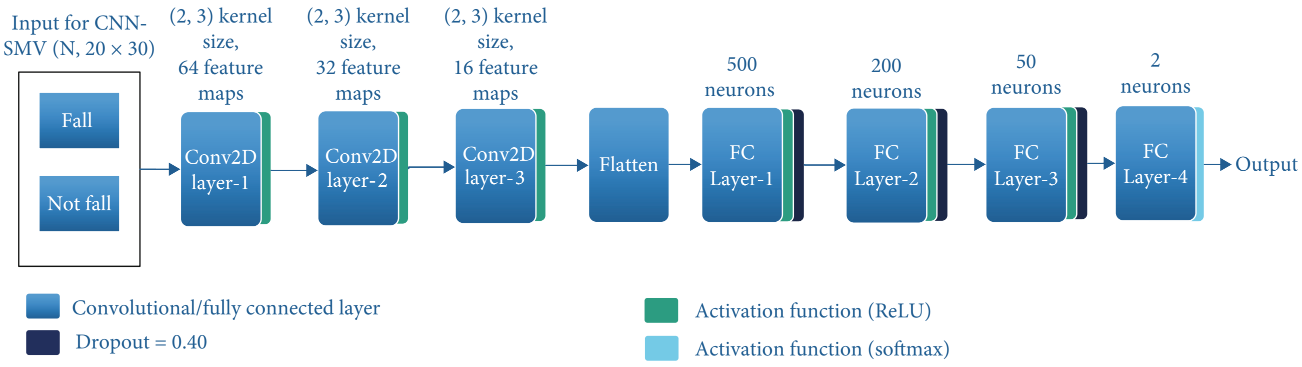
Input for CNN-SMV (N, 20 × 30)
Fall
Not fall
(2, 3) kernel size, 64 feature maps
Conv2D layer-1
(2, 3) kernel size, 32 feature maps
Conv2D layer-2
(2, 3) kernel size, 16 feature maps
Conv2D layer-3
Flatten
500 neurons
FC Layer-1
200 neurons
FC Layer-2
50 neurons
FC Layer-3
2 neurons
FC Layer-4
Output
Convolutional/fully connected layer
Dropout = 0.40
Activation function (ReLU)
Activation function (softmax)

Figure 9: Convolutional neural network architecture.

3.3. Convolutional Neural Network Design. To compare the performance of the proposed rule-based algorithm with machine learning (ML), we use a convolutional neural network (CNN), of which the architecture is shown in Figure 9. The CNN has shown tremendous performance in various classification problems such as image classification [74], modulation classification [75], and wireless technology classification [76–78]. In this work, the CNN is trained with three publicly available fall datasets presented in [79–81]. These datasets consist of the accelerometer data in the form of X, Y, and Z acceleration values, which were captured on the human body. The proposed rule-based algorithm identifies every event as either "Fall" or "Not Fall." In order to have a fair one-to-one comparison, the CNN is trained with 70 of each dataset that is categorising the events in a similar way. All the data classes corresponding to falls are combined in the "Fall" category, and all the activities of daily living (ADL) classes are combined in the "Not Fall" category. Therefore, the CNN also identifies the "Fall" and "Not Fall" events in a comparable way.

For the input to the CNN, we consider the SMV values of the raw X, Y, and Z accelerometer data computed according to formula (1). Furthermore, we consider a window size of $20 \times 30 = 600$ SMV values, which represents 3 in the time domain. The architecture of the CNN is composed of three 2D convolutional layers and four fully connected layers as shown in Figure 9. The last fully connected layer of the architecture is the softmax layer with two neurons, representing that the CNN is able to classify the two classes.

The total numbers of examples in KFall, SisFall1, SisFall2, and FallAIID are 5036, 4500, 4500, and 1798, respectively. Note that each example is of size (1×600), hence containing 600 SMV values computed according to formula (1) from (3×600) values of X, Y, and Z. Since we use Conv2D in the CNN architecture, we transformed each example from (1,600) to (1,20,30). For training and testing the classifier for each dataset, we divided the data into training, validation, and testing data sets with a split of 70/15/15.

For the optimiser, the Adam optimiser was used as it provided the best accuracy. In addition, *ReduceLROnPlateau* was used from the Tensorflow platform, because it helped in reducing the learning rate from 0.001 to 0.0001 when the validation loss stopped improving. In order to have a fine balance between overfitting and underfitting, a batch size of 512 and a dropout of 0.40 were used. The CNN classifier was trained on a NVIDIA GTX 1080Ti Graphics Card, which is available in our in-house Virtual Wall [82]. The classifier was implemented in an abstract level library Keras [83] with Tensorflow [84] as a back-end.

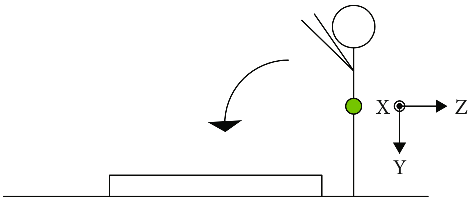

Figure 10: Graphical representation of the measurement setup: a person, approximately 1.8 m tall and wearing the wearable with integrated accelerometer around the waist, is falling forward on a mattress. The orientation of the accelerometer's axes is visualised.

4. Measurement and Analysis

4.1. Measurement Setup. To validate the algorithm, a set of simulations was performed based on the measurement setup in Figure 10. A person, approximately 1.8 m tall and wearing the device around his waist, falls on a mattress. Note that when a fall on a mattress is detected, a fall on a hard floor would certainly be detected as the acceleration at the moment of impact is certainly higher. During these measurements, the accelerometer measures negative values along the Y-axis, as shown in Figure 10. The MCU reads the accelerometer samples at a sample interval of 200 Hz and continuously transfers them to a PC used for data collection. Note that this is only necessary for the analysis and is not required in the actual application. The accelerometer was initialised to operate within a range of $\pm 4g$ and the integrated low-pass filtering was selected. Before sampling, inline accelerometer calibration [85] was performed according to the start-up procedure described by the manufacturer.

4.2. Fall. Multiple simulations are performed, to obtain the best sample rate for both detection branches, as well as the optimal buffer size and threshold values. The first parameter is the sample rate. Research proposed in [29, 32, 33] suggests a sample interval of 20 ms. However, taking into account

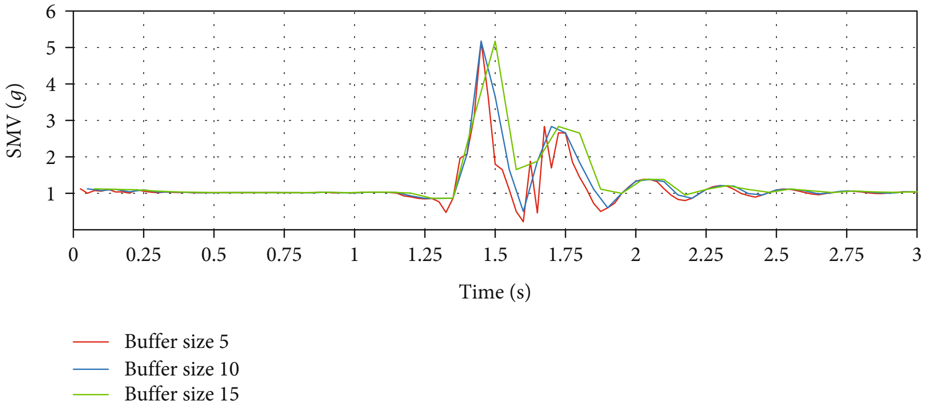

Figure 11: Maximum filtered SMV data for a buffer size of 10 samples of the fall measured in Figure 7. The results approach the theoretical fall profile.

that the peak of the fall only exists for approximately 50, we decided to sample at a higher interval of 5.

In Section 3.2, Figure 7 illustrates the measured fall at a sample interval of 5 ms or a sample rate of 200. The peak consisted of multiple samples above 3 g, which clearly illustrated that the peak cannot be missed at this sample rate.

The next parameter is the buffer size of the filter that determines the maximum SMV. A buffer size of 5 will lower the detection time but will increase the noise of the measurement. Utilising a buffer size of 15 will slow down the detection time but has a large influence on the noise filtering. A buffer size of 10 clearly provides the best result, approaching the theoretical fall profile as illustrated earlier in Figure 7, preserving the important information while at the same time reducing noise in the measurement. The result of filtering the measurement accordingly is displayed in Figure 11.

The last parameter is the threshold value that triggers an alert. From research performed in [70], we know that, while ascending or descending stairs, the SMV will never reach a peak as high as 1.7 g. Note that in nursing homes the elderly will generally not even use staircases.

Measurements were also performed to assess the SMV values when an elderly person falls back in a seat. The measurement setup was as described before, but now, the test person was standing in front of a seat, “fell” into the seat and stood up again. Figure 12 illustrates five measured “sit events” sampled at a rate of 200 Hz. As is shown in the graphs, the sit events did not cause any accelerations above 1.7 g whereas potential harmful fall events easily cause accelerations above a threshold of 2.5 g. This threshold value is chosen at a level that guarantees detection of dangerous fall events, while at the same time avoiding false alerts due to other harmless events.

After determining the optimal parameter values, we combined the acceleration and orientation branches. Figure 13 illustrates the complete algorithm applied to a fall measured at a sample interval of 5 ms. It can be seen that around 1.4, the acceleration alert flag is set. When taking a closer look, this flag is set at the peak of the fall, which proves the effectiveness of the acceleration detection. After a delay of 1.1 s, this fall is confirmed by the difference in spatial orientation, as can be seen in Figure 8. The confirmed fall alert will be sent at 2.5 s.

4.3. Validation with Open-Source Dataset. To verify the algorithm, we have selected three online fall detection databases. The first dataset is the SisFall dataset [79]. This dataset was created in 2017 and contains accelerometer data of two different accelerometer and gyroscope data from units mounted at the waist level. This dataset was created with 38 subjects ranging from young to old; the adults/elderly (>60 year) distribution is 23/15. Furthermore, it contains 34 ADL/falls that were repeated 1 or 5 times. In total, this dataset contains two times 4500 ADL/fall events.

The second dataset is the FallAIID dataset [80], which contains accelerometer, gyroscope, magnetometer, temperature, and barometer data measured around the waist, neck, and right hand. The 15 test subjects range from young to elderly adults, with 44/35 ADLs/falls at variable times per ADL/Fall. In total, this dataset contains 4760 ADL/Fall events. The third dataset, KFall [81], contains accelerometer, gyroscope, and magnetometer data captured also at the waist level. It is based on 32 young male test subjects performing 36 ADLs/falls variable times per ADL/fall. In total, this dataset contains 5075 ADL/fall events. A summary of the test subjects and ADL/Falls per dataset can be found in Tables 1 and 2, respectively.

Since the data in these datasets is captured with different accelerometers and since test subjects are falling in a different way with sometimes less representative falls, some parameter values of the proposed rule-based algorithm need to be adjusted. The first adjustment is the amplitude threshold. Since some measured ADL/Fall datasets contain more noise than others, it is important to set the threshold level above the noise level in order to suppress false alerts. To find the best suitable threshold, the histogram gives the maximum value of an ADL/fall event for each database, as can be seen in Figure 14. From this figure, it can be concluded that a threshold between 2 and 3 g suppresses most noise and will result in good performance of the algorithm.

Figure 15 illustrates the results of the proposed algorithm for different amplitude thresholds ranging from 1.75

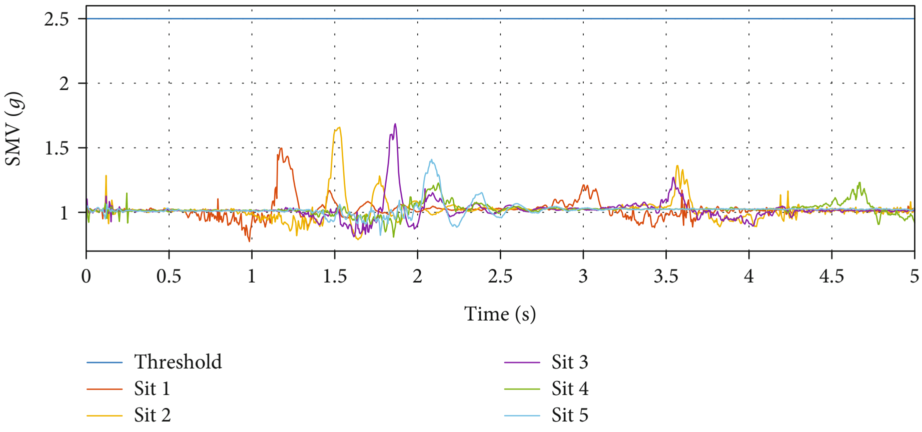

Figure 12: Graphical representation of five sit events measured at a sample interval of 5 ms.

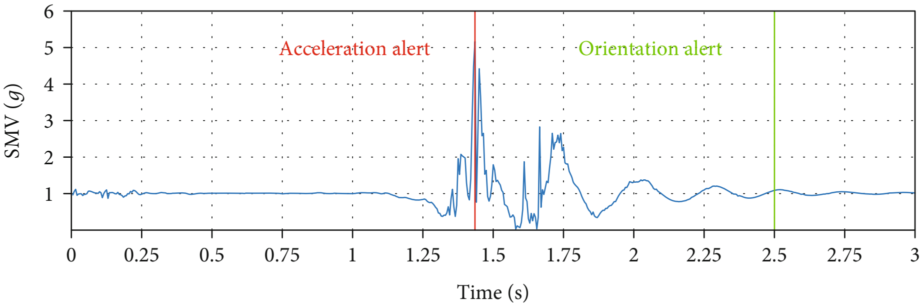

Figure 13: Results of the final algorithm applied to the fall from Figure 7, which was measured at a sample interval of 5. The red vertical line represents the acceleration alert and the green vertical line the orientation alert. Since both alerts are set, this is a confirmed fall event.

Table 1: Summary of the test subjects used in three datasets.

	Subjects	Age (yrs)	Weight (kg)	Height (cm)	Gender (M/F)
SisFall (2017)	38	(22–70)	(51-142.9)	(152-188)	19/19
FALLAIID (2021)	15	(21–53)	(48–85)	(158-187)	8/7
KFall (2021)	32	(21–29)	(60–79)	(168-180)	32/0

Table 2: Summary of the ADL/falls per dataset.

	Position	Times repeated	ADL/falls	Total events
SisFall (2017)	Waist	1 or 5	19/15	4500x2
FALLAIID (2021)	Waist, neck, right hand	Variable	44/35	4760
KFall (2021)	Waist	Variable	21/15	5075

to 3.5 g in steps of 0.25 g, with a constant timeslot of 1 and an orientation threshold of 45°. It can be concluded that a threshold of 2.5 g gives the best F_1-score for each dataset, proving that the previously chosen threshold performs well.

Second, we considered the orientation threshold. During extensive testing, it appeared as if some test subjects did not wear the accelerometer unit tight enough to the body. In these situations, where a person fell when trying to sit or stand up from a chair, the recorded orientation differences in the database are smaller than what is realistically possible. In order to compensate for this underestimation, the threshold was lowered to 45° instead of 60°.

A final adjustment is the timeslot. To have a clear distinction between a fall and for example a forward jump, the timeslot is increased. In this way, the algorithm is prevented from calculating the orientation difference during the fall or jumping. Table 3 gives a short overview of the parameters that were changed for each dataset. The amplitude threshold differs per dataset since this gives us the best result for each dataset. Setting this threshold to the same value results in a minor difference in F_1-score, as can be seen in Figure 15.

Table 4 represents the accuracy, precision, sensitivity, and F_1-score of the developed algorithm applied to all three

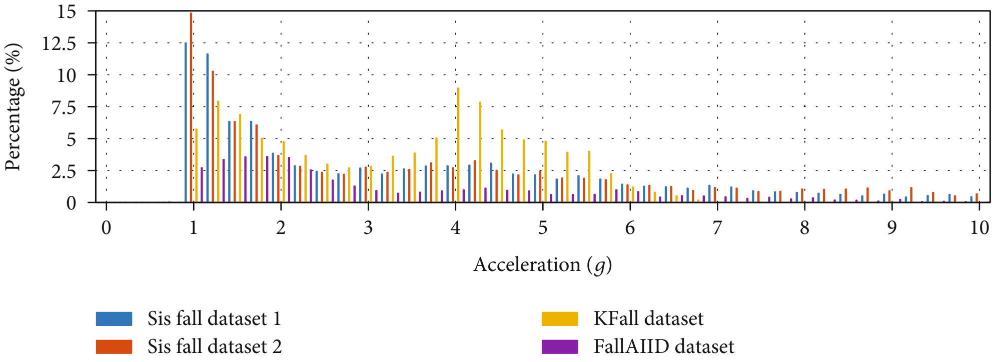

FIGURE 14: Calculated histogram of the max values of each event in the dataset.

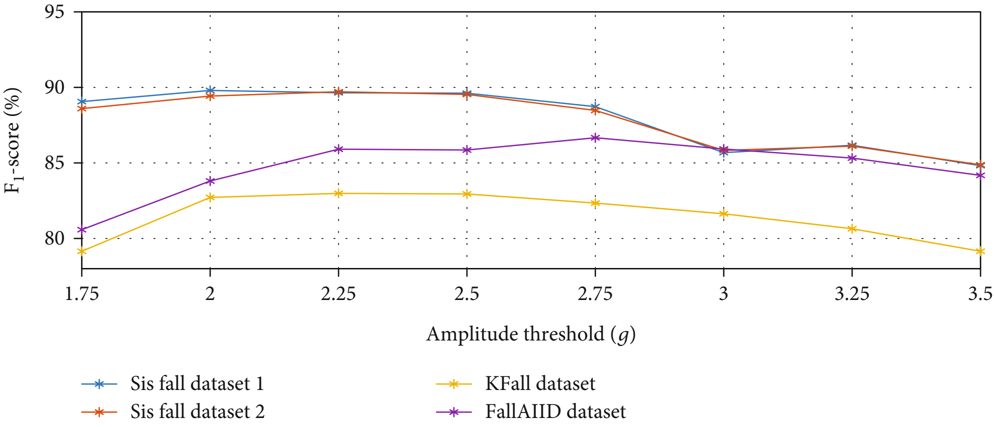

FIGURE 15: Results of the amplitude range simulations of the proposed algorithm.

TABLE 3: Summary of the adjusted parameters for each dataset.

	Paper	SisFall (2017)	FALLAIID (2021)	KFall (2021)
Sample rate	5 ms	5 ms	4.2 ms	10
Amplitude threshold	2.5g	2 g	2.75 g	2.5 g
Orientation threshold	60°	45°	45°	45°
Timeslot	0.25 s	1	1	1

TABLE 4: Summary of the obtained accuracy, precision, sensitivity, and F_1-score for the proposed algorithm per datasets.

	SisFall (2017)		FALLAIID (2021)	KFall (2021)
Accuracy	91.58%	91.27%	92.65%	84.89%
Precision	87.01%	86.61%	81.87%	85.35%
Sensitivity	92.77%	92.44%	92.06%	80.66%
F_1-score	89.80%	89.43%	86.66%	82.94%

TABLE 5: Summary of the CNN algorithm for each dataset. The CNN was trained on 70% data, validated on 15% and tested on 15% of each dataset.

	SisFall (2017)		FALLAIID (2021)	KFall (2021)
Accuracy	95.85%	96.07%	94.44%	93.44%
Precision	95.38%	93.28%	84.48%	93.43%
Sensitivity	94.34%	96.98%	89.09%	93.30%
F_1-score	94.86%	95.09%	86.72%	93.36%

datasets. It results in an accuracy in the range of 84.89% to 92.65%, representing the percentage of correct decisions. A precision in a range of 81.87% to 87.01% is obtained, representing the ratio of true fall detections over all fall detections. A sensitivity in the range of 80.66% to 92.77% is observed, corresponding to the ratio of correct fall detections to the number of actual falls events. Based on these values, the F_1-scores are in a range of 82.94% to 89.8%, confirming the reliability of the algorithm. One must take into account that these datasets also contain ADLs such as jogging and jumping. These can trigger false alerts in some situations, even though they do not occur often in the lives of the elderly.

Table 5 also shows the performance of the CNN algorithm in terms of accuracy, precision, sensitivity and F_1-score. The algorithm was trained on 70% training data and tested and validated on the 15% data for each dataset. There can be stated that the results are higher than the results from the proposed rule based algorithm (Table 4), but at a much higher power consumption. Furthermore, it can be stated that CNN is very data dependent. When the CNN is trained on both SisFall datasets and KFall and later validated on FALLAIID, we get an accuracy of 90.32%, a

Table 6: Proposed scenario for a day of an elderly person in a nursing home, based on information from a local home.

Time	Activity	Microcontroller mode	Accelerometer mode
8:00-8:30	Wake-up, getting dressed and walking towards the kitchen for breakfast	Active	Active
8:30-9:30	Having breakfast and socialising with other residents	Deep sleep	Active
9:30-935	Walking towards the garden, room or recreation centre	Active	Active
9:35-12:00	Enjoying the moment	Deep sleep	Active
12:00-1205	Walking towards the kitchen for lunch	Active	Active
12:05-13:05	Having lunch and socialising with other residents	Deep sleep	Active
13:05-13:10	Walking towards room	Active	Active
13:10-14:30	Sleeping	Deep sleep	Active
14:30-14:35	Walking towards the garden, room or recreation centre	Active	Active
14:35-1730	Enjoying the moment	Deep sleep	Active
17:30-1735	Walking towards the kitchen for dinner	Active	Active
17:35-18:35	Having dinner and socialising with the other residents	Deep sleep	Active
18:35-18:40	Walking back to the room	Active	Active
18:40-21:00	Sitting or lying down in bed watching TV	Deep sleep	Active
21:00-22:30	Sleeping (time needed to detect elderly person is sleeping)	Deep sleep	Active
22:30-8:00	Sleeping	Deep sleep	Low power

precision of 73.40%, a sensitivity of 87.24%, and an F_1-score of 79.72%.

However, the CNN also exhibits some drawbacks: (a) it requires training, and (b) it consumes more resources while executing on embedded platforms as compared to the rule-based algorithm. Due to the advancement of ML on embedded platforms, the second problem can be alleviated, since now procedures exist for generating lightweight models including Tensorflow libraries such as Tensorflow Lite and TensorRT. The aim of this paper is only to identify "Fall" and "No Fall" events, and in this case, the proposed algorithm performs similar than the CNN algorithm because (a) it provides higher accuracy, (b) there is no need for training, and (c) it yields a lightweight algorithm. However, if the goal is to identify other "Fall" and "No Fall" events such as walking, standing, and picking up objects, then developing a rule-based algorithm for such a case would be extremely difficult or rather impossible. In that case, ML would be a nice alternative, because it does not require development of such rules but rather a CNN must be trained by a labelled data set and a similar performance as for a rule-based algorithm can be expected.

5. Power Management

In a wearable device, the battery lifetime is an important property. When the elderly person is walking around, the MCU and the accelerometer will be in *active mode* and consume power to perform the programmed tasks. Utilising the *no-motion interrupt* of the accelerometer, the MCU is put in a *deep sleep mode*, which dramatically decreases the consumed power. To wake up the MCU when the person is walking again, the *High-g interrupt* of the accelerometer is used. To lower the power consumption even more, the accelerometer will be put in a *low-power mode*, for example, when the elderly person goes to bed and is sleeping for multiple hours. Based on information from a local nursing home for the elderly person, the following scenario is proposed in Table 6.

When both MCU and the accelerometer are in active mode, there is an energy consumption corresponding to an hourly charge of 0.45 mAh. Putting the MCU in deep sleep mode decreases the daily required battery charge to 0.13 mAh. When the MCU is in deep sleep mode and the accelerometer is in low-power mode, the total required charge is 7.9 μAh. Taking these consumptions into account, the total required charge for this scenario per day is 2.27 mAh.

Furthermore, the wearable is programmed to send a "standby" signal every 15 minutes in order to inform the system that the wearable is still active. One such message consists of five subsequent advertisement packets in order to assure proper detection by the system. The 5 packets are sent at a TX-level of 1, which results in a series of peak currents of 9 mA during approximately 20 ms for each message [65]. In a worst-case scenario, the elderly person falls once a day, which results in another five current peaks of 9 mA during 20 ms for each message.

Taking all of these parameters into account and using a 3 V CR2032 battery with a capacity of 230 mAh, this results in a battery lifetime of 100 days. The CR2032 was chosen for reasons of compactness and ease of use during development. In this way, we meet the third and fourth design requirements. A larger (rechargeable) cell can always be used if a longer autonomy is preferred. As an example, applying a CR2477N 3V Lithium Battery with a capacity of 950 mAh would extend the autonomy to more than 414 days maximum, if self-discharge of the battery is not taken into account.

6. Conclusion

This paper describes the design of a wearable fall detection sensor for the elderly based on BLE wireless communication, providing excellent room coverage at low energy consumption. The wearable relies on a three-axis accelerometer to find the peak of the fall as well as the change in spatial orientation. This change in spatial orientation is used as a fall confirmation signal. A false detection resulting in a confirmed fall alert is virtually impossible with this approach.

The system is designed, implemented, and validated by means of a measurement campaign. In order to assess sensitivity and selectivity, not only fall events were measured but also other harmless conditions causing fairly large accelerations, such as "falling" into a seat.

Based on the measurement results, realistic parameters were chosen for the algorithms, of which the most important are the threshold values of 2.5 g and 60° for the acceleration and orientation change, respectively. A sample rate of 5 ms was chosen in order to surely capture all relevant details of each fall event.

A validation of the algorithm was performed using three open-source databases containing data for 85 persons, performing 18 835 ADL or fall activities. Based on these data, the algorithm obtained an F_1 score in a range of 82.94% to 89.8%, confirming its reliability and illustrating that the first and second design requirements are met. In addition, the proposed rule-based algorithm is compared to ML using the CNN. The results show that the CNN algorithm yields an F_1 performance in a range of 86.72% to 95.09%, which is comparable to the proposed rule-based algorithm. This way the first and second design requirements are met.

The proposed wearable provides reliable fall detection, without false acceleration alerts (level 1). The orientation confirmation system provides a second, independent variable, making false (level 2) confirmed positives extremely unlikely. The proposed unobtrusive wearable system fits in a $32 \times 23 \times 10$ mm package and autonomously performs measurements as well as communication employing state of the art wireless technology. The system is energy efficient and achieves an autonomy of 100 days on a standard CR2032 coin-cell battery. In this way, the third and fourth design requirements are also met.

Data Availability

In this research, we measured our own data to develop the algorithm (contact nick.deraeve@ugent.be) and used the data from three publicly available datasets to validate our algorithm.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This work was supported in part by the Research Foundation Flanders (FWO) through the "MUlti-SErvice WIreless NETwork," FWO/FRS Excellence of Science (EOS) Project and the Erasmus+ programme of the European Union through the "Capacity building for Digital Health Monitoring and Care Systems in Asia (DigiHealth-Asia), 2021-2024" project.

References

[1] World Health Organization, "Number of people over 60 years set to double by 2050; major societal changes required," January 2022, https://www.who.int/mediacentre/news/releases/2015/older-persons-day/en/.

[2] World Health Organization, "World 397 health organization global report on falls prevention in older age 2007," January 2022, https://extranet.who.int/agefriendlyworld/wp-content/uploads/2014/06/WHo-Global-report-on-falls-prevention-in-older-age.pdf.

[3] World Health Organization, "Falls prevention in older age," January 2022, https://www.who.int/ageing/projects/falls_prevention_older_age/en/.

[4] S. R. Lord, A. Tiedemann, K. Chapman et al., "The effect of an individualized fall prevention program on fall risk and falls in older people: a randomized, controlled trial," *Journal of the American Geriatrics Society*, vol. 53, no. 8, pp. 1296–1304, 2005.

[5] B. Røyset, B. A. Talseth-Palmer, S. Lydersen, and P. G. Farup, "Effects of a fall prevention program in elderly: a pragmatic observational study in two orthopedic departments," *Clinical Interventions in Aging*, vol. 14, no. 1, pp. 145–154, 2019.

[6] C. S. Florence, G. Bergen, A. Atherly, E. Burns, J. Stevens, and C. Drake, "Medical costs of fatal and nonfatal falls in older adults," *Journal of the American Geriatrics Society*, vol. 66, no. 4, pp. 693–698, 2018.

[7] M. Woolley, *Bluetooth Core Specification v5.1*, Bluetooth Special Interest Group, 2019.

[8] A. Makhlouf, I. Boudouane, N. Saadia, and A. Ramdane Cherif, "Ambient assistance service for fall and heart problem detection," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1527–1546, 2019.

[9] L. Ren and Y. Peng, "Research of fall detection and fall prevention technologies: a systematic review," *IEEE Access*, vol. 7, pp. 77702–77722, 2019.

[10] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.

[11] X. Wang, J. Ellul, and G. Azzopardi, "Elderly fall detection systems: a literature survey," *Frontiers in Robotics and AI*, vol. 7, p. 71, 2020.

[12] G. Koshmak, A. Loutfi, and M. Linden, "Challenges and issues in multisensor fusion approach for fall detection: review paper," *Journal of Sensors*, vol. 2016, Article ID 6931789, 12 pages, 2016.

[13] T. Xu, Y. Zhou, and J. Zhu, "New advances and challenges of fall detection systems: a survey," *Applied Sciences*, vol. 8, no. 3, p. 418, 2018.

[14] S. Nooruddin, M. M. Islam, F. A. Sharna, H. Alhetari, and M. N. Kabir, "Sensor-based fall detection systems: a review," *Journal of Ambient Intelligence and Humanized Computing*, 2021.

[15] C. Vishnu, R. Datla, D. Roy, S. Babu, and C. K. Mohan, "Human fall detection in surveillance videos using fall motion vector modeling," *IEEE Sensors Journal*, vol. 21, no. 15, pp. 17162–17170, 2021.

[16] J. Gutiérrez, V. Rodríguez, and S. Martin, "Comprehensive review of vision-based fall detection systems," *Sensors*, vol. 21, no. 3, p. 947, 2021.

[17] C. Kong, L. Chen, Z. Wang, Y. Chen, L. Meng, and H. Tomiyama, “Robust self-adaptation fall-detection system based on camera height,” *Sensors*, vol. 19, no. 17, p. 3768, 2019.

[18] W.-Y. Cai, J.-H. Guo, M.-Y. Zhang, Z.-X. Ruan, X.-C. Zheng, and S.-S. Lv, “GBDT-based fall detection with comprehensive data from posture sensor and human skeleton extraction,” *Journal of Healthcare Engineering*, vol. 2020, Article ID 32676176, 15 pages, 2020.

[19] J. R. Torres Neto, G. P. Rocha Filho, L. Y. Mano, L. A. Villas, and J. Ueyama, “Exploiting offloading in IoT-based microfog: experiments with face recognition and fall detection,” *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 2786837, 13 pages, 2019.

[20] E. Minguez and M. Faundez-Zanuy, “Low cost fall detection based on cortex m4,” in *2019 42nd international conference on telecommunications and signal processing (TSP)*, pp. 36–40, Budapest, Hungary, 2019.

[21] M. Alwan, P. J. Rajendran, S. Kell et al., “A smart and passive floor-vibration based fall detector for elderly,” in *2006 2nd International Conference on Information Communication Technologies*, pp. 1003–1007, Damascus, Syria, 2006.

[22] G. Feng, J. Mai, Z. Ban, X. Guo, and G. Wang, “Floor pressure imaging for fall detection with fiber-optic sensors,” *IEEE Pervasive Computing*, vol. 15, no. 2, pp. 40–47, 2016.

[23] A. M. Tabar, A. Keshavarz, and H. Aghajan, “Smart home care network using sensor fusion and distributed vision-based reasoning,” in *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks, ser. VSSN’06*, pp. 145–154, New York, NY, USA, 2006.

[24] D. Martelli, F. Artoni, V. Monaco, A. M. Sabatini, and S. Micera, “Pre-impact fall detection: optimal sensor positioning based on a machine learning paradigm,” *PLoS One*, vol. 9, no. 3, article e92037, 2014.

[25] T. Mauldin, M. Canby, V. Metsis, A. Ngu, and C. Rivera, “SmartFall: a smartwatch-based fall detection system using deep learning,” *Sensors*, vol. 18, no. 10, p. 3363, 2018.

[26] A. Shahzad and K. Kim, “FallDroid: an automated smartphone-based fall detection system using multiple kernel learning,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 35–44, 2019.

[27] P. Kostopoulos, T. Nunes, K. Salvi, M. Deriaz, and J. Torrent, “Increased Fall Detection Accuracy in an Accelerometer-Based Algorithm Considering Residual Movement,” in *Proceedings of the International Conference on Pattern Recognition Applications and Methods*, pp. 30–36, 2015.

[28] G. B. Huq, J. Basilakis, and A. Maeder, “Evaluation of tri-axial accelerometery data of falls for elderly through smart phone,” in *Proceedings of the Australasian Computer Science Week Multiconference on - ACSW 16*, pp. 1–10, 2016.

[29] S. Khojasteh, J. Villar, C. Chira, V. González, and E. de la Cal, “Improving fall detection using an on-wrist wearable accelerometer,” *Sensors*, vol. 18, no. 5, p. 1350, 2018.

[30] Y. Su, D. Liu, and Y. Wu, “A multi-sensor based pre-impact fall detection system with a hierarchical classifier,” in *2016 9th international congress on image and signal processing, Bio-Medical engineering and informatics (CISP-BMEI)*, pp. 1727–1731, Datong, China, 2016.

[31] D. Ajerla, S. Mahfuz, and F. Zulkernine, “A real-time patient monitoring framework for fall detection,” *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 9507938, 13 pages, 2019.

[32] A. Shahzad, S. Ko, S. Lee, J.-A. Lee, and K. Kim, “Quantitative assessment of balance impairment for fall-risk estimation using wearable triaxial accelerometer,” *IEEE Sensors Journal*, vol. 17, no. 20, pp. 6743–6751, 2017.

[33] Y. Liu, S. J. Redmond, T. Shany et al., “Validation of an accelerometer-based fall prediction model,” in *2014 36th annual international conference of the IEEE engineering in medicine and biology society*, pp. 4531–4534, Chicago, IL, USA, 2014.

[34] F. Luna-Perejón, M. J. Domínguez-Morales, and A. Civit-Balcells, “Wearable fall detector using recurrent neural networks,” *Sensors*, vol. 19, no. 22, p. 4885, 2019.

[35] E. Casilari, R. Lora-Rivera, and F. García-Lagos, “A study on the application of convolutional neural networks to fall detection evaluated with multiple public datasets,” *Sensors*, vol. 20, no. 5, p. 1466, 2020.

[36] L. Palmerini, J. Klenk, C. Becker, and L. ChiariAccelerometer-based fall detection using machine learning: training and testing on real-world falls,” *Sensors*, vol. 20, no. 22, p. 6479, 2020.

[37] G. Santos, P. Endo, K. Monteiro, E. Rocha, I. Silva, and T. Lynn, “Accelerometer-based human fall detection using convolutional neural networks,” *Sensors*, vol. 19, no. 7, p. 1644, 2019.

[38] N. Liu, D. Zhang, Z. Su, and T. Wang, “Preimpact fall detection for elderly based on fractional domain,” *Mathematical Problems in Engineering*, vol. 2021, Article ID 6661034, 17 pages, 2021.

[39] S. B. Kwon, J.-H. Park, C. Kwon, H. J. Kong, J. Y. Hwang, and H. C. Kim, “An energy-efficient algorithm for classification of fall types using a wearable sensor,” *IEEE Access*, vol. 7, pp. 31321–31329, 2019.

[40] C. Gilly and V. A. Zeithaml, “The elderly consumer and adoption of technologies,” *Journal of Consumer Research*, vol. 12, no. 3, p. 353, 1985.

[41] K.-C. Yang and P.-H. Shih, “Cognitive age in technology acceptance: at what age are people ready to adopt and continuously use fashionable products?,” *Telematics and Informatics*, vol. 51, article 101400, 2020.

[42] A. Ramachandran and A. Karuppiah, “A survey on recent advances in wearable fall detection systems,” *BioMed Research International*, vol. 2020, Article ID 2167160, 17 pages, 2020.

[43] T. Tamura, T. Yoshimura, M. Sekine, M. Uchida, and O. Tanaka, “A wearable airbag to prevent fall injuries,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 6, pp. 910–914, 2009.

[44] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy, “Wearable sensors for reliable fall detection,” in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pp. 3551–3554, Shanghai, China, 2005.

[45] F. Wu, H. Zhao, Y. Zhao, and H. Zhong, “Development of a wearable-sensor-based fall detection system,” *International Journal of Telemedicine and Applications*, vol. 2015, Article ID 25784933, 11 pages, 2015.

[46] T. Nguyen Gia, V. K. Sarker, I. Tcarenko et al., “Energy efficient wearable sensor node for IoT-based fall detection systems,” *Microprocessors and Microsystems*, vol. 56, pp. 34–46, 2018.

[47] P. Fino, C. Frames, and T. Lockhart, “Classifying step and spin turns using wireless gyroscopes and implications for fall risk assessments,” *Sensors*, vol. 15, no. 5, pp. 10676–10685, 2015.

[48] A. Bourke and G. Lyons, “A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor,” *Medical Engineering & Physics*, vol. 30, no. 1, pp. 84–90, 2008.

[49] O. Almeida, M. Zhang, and J.-C. Liu, “Dynamic fall detection and pace measurement in walking sticks,” in *2007 Joint workshop on high confidence medical devices, software, and systems and medical device plug-and-play interoperability (HCMDSS-MDPnP 2007)*, pp. 204–206, Boston, MA, USA, 2007.

[50] E. Casilari, M. Álvarez-Marco, and F. García-Lagos, “A study of the use of gyroscope measurements in wearable fall detection systems,” *Symmetry*, vol. 12, no. 4, p. 649, 2020.

[51] S.-L. Hsieh, C.-C. Chen, S.-H. Wu, and T.-W. Yue, “A wrist -worn fall detection system using accelerometers and gyroscopes,” in *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control*, pp. 518–523, Miami, FL, USA, 2014.

[52] F.-T. Wang, H.-L. Chan, M.-H. Hsu, C.-K. Lin, P.-K. Chao, and Y.-J. Chang, “Threshold-based fall detection using a hybrid of tri-axial accelerometer and gyroscope,” *Physiological Measurement*, vol. 39, no. 10, article 105002, 2018.

[53] Lifeline, “Medical alert systems for seniors with fall detection,” January 2022, https://www.lifeline.ca/en/about-us/fall554detection-technology-autoalert/.

[54] Apple, “Use fall detection with apple watch,” January 2022, https://support.apple.com/en-us/HT208944.

[55] “Owlytics Healthcare,” January 2022, https://www.owlytics.com/.

[56] Vayyar Home, “Vayyar home sensors,” January 2022, https://vayyarhome.com/.

[57] J. G. Argañarás, Y. T. Wong, R. Begg, and N. C. Karmakar, “State-of-the-art wearable sensors and possibilities for radar in fall prevention,” *Sensors*, vol. 21, no. 20, p. 6836, 2021.

[58] “HipHope technologies,” January 2022, https://www.hip-hope.com/.

[59] Bluetooth Special Interest Group, “Bluetooth mesh glossary of terms,” January 2022, https://www.bluetooth.com/learn-about-bluetooth/bluetooth-technology/topology-options/le-mesh/mesh-glossary/#proxy.

[60] Bluetooth Special Interest Group, “An introduction to the bluetooth mesh proxy function,” January 2022, https://www.bluetooth.com/bluetooth-resources/bluetooth-mesh-proxy-kit/.

[61] N. S. Suriani, F. A. N. Rashid, and N. Y. Yunos, “Optimal accelerometer placement for fall detection of rehabilitation patients,” *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 10, pp. 25–29, 2018.

[62] A. Özdemir, “An analysis on sensor locations of the human body for wearable fall detection devices: principles and practice,” *Sensors*, vol. 16, no. 8, p. 1161, 2016.

[63] P. Bulic, G. Kojek, and A. Biasizzo, “Data transmission efficiency in bluetooth low energy versions,” *Sensors*, vol. 19, no. 17, p. 3746, 2019.

[64] Bluetooth Special Interest Group, “Bluetooth generic attributes,” January 2022, https://www.bluetooth.com/specifications/gatt/.

[65] Silicon Labratories, “Bgm13p blue gecko bluetooth module data sheet,” January 2022, https://www.silabs.com/documents/public/data-sheets/bgm13p-datasheet.pdf.

[66] ARMJanuary 2022, https://www.arm.com/products/silicon-ip-cpu.

[67] “TAG connect,” January 2022, http://www.tag-connect.com/.

[68] Silicon Laboratories, “Programming internal flash over the serialwire debug interface,” January 2022, https://www.silabs.com/documents/public/application-notes/an0062.pdf.

[69] Bosch, “Accelerometer sensor bma280,” January 2022, https://www.bosch-sensortec.com/products/motion-sensors/accelerometers/bma280.html.

[70] T. Xu, W. Sun, S. Lu, K. M. Ma, and X. Wang, “The real-time elderly fall posture identifying scheme with wearable sensors,” *International Journal of Distributed Sensor Networks*, vol. 15, no. 11, 2019.

[71] E. C. Ifeachor and B. W. Jervis, *Digital Signal Processing: A Practical Approach*, Pearson Education, 2nd edition, 2002.

[72] M. H. Hayes, *Schaum’s Outline of Digital Signal Processing*, McGraw-Hill, Inc., 1st edition, 1998.

[73] C. L. Karmaker, “Determination of optimum smoothing constant of single exponential smoothing method: a case study,” *International Journal of Research in Industrial Engineering*, vol. 6, no. 3, pp. 184–192, 2017.

[74] D. Ciregan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3642–3649, Providence, RI, USA, 2012.

[75] T. J. O’Shea, J. Corgan, and T. C. Clancy, “Convolutional radio modulation recognition networks,” in *International Conference on Engineering Applications of Neural Networks*, pp. 213–226, 2016.

[76] J. Fontaine, E. Fonseca, A. Shahid et al., “Towards low-complexity wireless technology classification across multiple environments,” *Ad Hoc Networks*, vol. 91, article 101881, 2019.

[77] A. Shahid, J. Fontaine, M. Camelo et al., “A convolutional neural network approach for classification of lpwan technologies: SigFox, loRA and IEEE 802.15.4g,” in *2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1–8, Boston, MA, USA, 2019.

[78] M. Camelo, A. Shahid, J. Fontaine et al., “A semi-supervised learning approach towards automatic wireless technology recognition,” in *2019 IEEE international symposium on dynamic Spectrum access networks (DySPAN)*, pp. 1–10, Newark, NJ, USA, 2019.

[79] A. Sucerquia, J. D. López, and J. F. Vargas-Bonilla, “Sisfall: a fall and movement dataset,” *Sensors*, vol. 17, no. 12, p. 198, 2017.

[80] M. Saleh, M. Abbas, and R. B. L. Jeannes, “FallAllD: an open dataset of human falls and activities of daily living for classical and deep learning applications,” *IEEE Sensors Journal*, vol. 21, no. 2, pp. 1849–1858, 2021.

[81] X. Yu, J. Jang, and S. Xiong, “A large-scale open motion dataset (KFall) and benchmark algorithms for detecting pre-impact fall of the elderly using wearable inertial sensors,” *Frontiers in Aging Neuroscience*, vol. 13, 2021.

[82] “imec testbed,” January 2022, https://doc.ilabt.imec.be/ilabt/virtualwall/.

[83] “Keras,” January 2022, https://keras.io/.

[84] “Tensorflow,” January 2022, https://www.tensorflow.org/.

[85] Bosch, “Multiple acceleration sensors - inline calibration,” January 2022, https://www.bosch-sensortec.com/media/boschsensortec/downloads/application_notes_1/bst-mas-an030.pdf.