

A FPGA threshold-based fall detection algorithm for elderly fall monitoring with verilog

Pui Mun Lo, Azniza Abd Aziz

School of Electrical and Electronics Engineering, Universiti Sains Malaysia, Pulau Pinang, Malaysia

Article Info

Article history:

Received Nov 12, 2020
Revised Mar 2, 2021
Accepted Jul 26, 2021

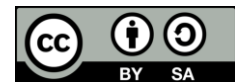
Keywords:

Fall detection
Fall like activities
FPGA
RTL
Thresholding

ABSTRACT

Fall is one of the leading causes of accidental or unintentional injury deaths worldwide due to serious injuries such as head traumas and hip fractures. As life expectancy improved, the rapid increase in aging population implied the need for the development of vital sign detector such as fall detector to help elderly in seeking for medical attention. Immediate rescue could prevent victims from the risk of suspension trauma and reduce the mortality rate among elderly population due to fall accident effectively. This paper presents the development of FPGA-based fall detection algorithm using a threshold-based analytical method. The proposed algorithm is to minimize the rate of false positive fall detection proposed from other researchers by including the non-fall events in the data analysis. Based on the performance evaluation, the proposed algorithm successfully achieved a sensitivity of 97.45% and specificity of 97.38%. The proposed algorithm was able to differentiate fall events and non-fall events effectively, except for fast lying and fall that ending with sitting position. The proposed algorithm shows a good result and the performance of the proposed algorithm can be further improved by using an additional gyroscope to detect the posture of the lower body part.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Azniza Abd Aziz
School of Electrical and Electronic Engineering
Universiti Sains Malaysia
Engineering Campus, Seberang Perai Selatan, Malaysia
Email: azniza@usm.my

1. INTRODUCTION

The average life expectancy in most of the countries has increased dramatically over the past few decades due to the vast improvements in healthcare services. In recent years, the aging population of Malaysia is estimated to be 2.2 million and tends to increase to 3.3 million in the year 2020 [1]. According to the world health organization (WHO), the aging population is expected to reach up to 22% of the world population in 2050 [2]. The rapid increase in aging population implied the need for vital sign detector application and assistive technology to enhance the functional needs of the elderly population who require special care. The invention of vital sign detector applications and assistive technology could ease the independent lifestyle of elderly people while keeping them safe and protected.

According to WHO, fall is one of the leading causes of accidental or unintentional injury deaths worldwide due to serious injuries such as head traumas and hip fractures. There are over 37.3 million severe injuries and 646,000 deaths due to fall being reported per annum [3]. Elderly and physically disabled people experience a higher risk of fall hazards when they are staying alone at home. Elderly falls are even exposed to the risk of fragile bones due to osteoporosis. The immediate rescue could prevent victims from the risk of suspension trauma and reduce the mortality rate effectively. Hence, a fall detector is required to aid the

elderly to seek for immediate rescue once falling accident has happened. The main challenge in developing a fall detection system is the effectiveness and accuracy of the algorithm in detecting a true positive falling accident from normal daily activities (NDA). Based on the current research, there are various type of fall detection approaches for elderly and people with disabilities, which includes wearable based, vision-based and ambient based approach [4]. These approaches analyse fall events using a simple threshold and machine learning methods. The miss rate due to fall-like activities is the common limitation found in the current researches of a sensor-based fall detection approach. The experimental results have proved that the rate of false-positive or false-negative fall detection is higher if only acceleration data was being considered. Although a vision-based approach could provide better performance in fall detection but the sensor-based approach is more preferable due to privacy invasion.

In this paper, a FPGA-based fall detection algorithm using a threshold-based analytical method is proposed. The proposed algorithm recognizes a fall event from NDA with the combination of accelerometer and gyroscope threshold data to avoid high rate of false-positive fall detection in acceleration mean. An FPGA is used as the core controller in fall detection to ensure effective parallel processing and fast response to data outputs from the sensor. Although the machine learning approach has higher performance, the threshold-based method is chosen to guarantee low implementation cost [5].

2. RELATED WORK

Fall detection approaches can be classified into sensor-based and vision-based approach. Sensor-based approach involves the use of ambient sensor and kinematic sensor. Ambient sensor-based approach is limited to indoor environment or a confined area. The system relies on the data from sensors that are installed in a specific area to detect falls. Meanwhile, kinematic sensor-based approach is applied to wearable device to measure human motion data. Thus, it is flexible to be used in any environment. The use of smartphone in fall detection is also considered as sensor-based approach. The analytical method of fall detection can be categorized into two fields: thresholding and machine learning method. Threshold-based fall detection uses a predefined value to detect falls. The system that works on the threshold-based method detects fall events by comparing the collected sensors data with the pre-set value. A fall accident is detected if the sensor reading exceeds the critical value defined in the system. The performance of the system is highly dependent on the predefined threshold value [6]. This method is commonly used in a sensor-based fall detection approach. In machine learning-based fall detection, the predictive model is expected to detect falls from a new dataset after training with the historic data [7]. It is commonly used in vision-based and ambient sensor-based approaches. Wearable sensor-based approach is the most commonly used method for fall detection system. It makes use of wearable devices and kinematic sensors to provide the relevant information for fall detection. It relies on the use of embedded sensors such as accelerometer and gyroscope to detect linear motion and posture of human body. Accelerometer and gyroscope data will be collected and transmitted to the microcontroller for data processing. The system will differentiate fall accident from NDA based on the change in acceleration of human body together with the change in angular velocity [8]. It is applicable for either indoor or outdoor environment and easy to wear.

Irene *et al.* [9] and Abdelhedi *et al.* [10] proposed a 3-axial accelerometer-based fall detection system which utilizes multiple predefined threshold values to detect falls. Tri *et al.* [11] employed the smartphone acceleration sensor for automatic fall detection. An additional long lie detection algorithm is used to improve the accuracy of the system. However, the system experienced a high risk of false-positive fall detection due to fall-like activities. Huynh *et al.* [12] found that this efficacy can be improved by adding a 3-axis gyroscope to the accelerometer-based wearable fall detector. The additional information from the gyroscope (change in angular velocity) enhances the sensitivity of the device by providing further delineation of a fall incident. However, further enhancement is required for this system to detect the direction where a fall accident has happened, and this could aid medical diagnosis effectively. Vetsandonphong [13] developed an Arduino based fall detection wearable device by coupling accelerometer with gyroscope. The wearable device is attached to human body using body strap. A fall event is determined based on the acceleration and row reading from the gyroscope. The wearable device is mounted on the upper trunk of user's body to minimized false positive fall detection with 95% sensitivity and 90% respectively.

An IoT-based fall detection system for elderly people is presented by Yacchirema *et al.* [14] using sensor network and big data. The proposed system involves a wearable device, low-power wireless sensor networks, cloud computing and big data. A 3D-axis accelerometer is embedded into a 6LoWPAN wearable device and placed at elderlies' waist to collect real time motion data. If a fall event occurs, the system remotely alerts the emergency center via QoS mechanisms. Data processing and analysis are done by using machine learning processing techniques based on decision trees-based big data model that running on a smart IoT gateway. The limitation of the proposed system is that it only applicable for indoor environment.

Torres *et al.* [15] introduced a chest mounted EnOcean fall detection wearable sensor for ultra-low power networks that capable to interact with a smart home system. The wearable sensor detects fall incident by using data fusion of accelerometer and gyroscope. A simple threshold algorithm is developed and fall detection will be based on the linear acceleration, aggregate rotational rate, angles inclination on X and Z planes. The algorithm is capable to detect forward, backward and lateral falls effectively. The system achieved a sensitivity of 96% in detecting a fall and specificity of 100% while performing NDA movements. However, fall accidents such as stair fall that does not end up with lying stage is undetectable in this algorithm.

Miguel *et al.* [16] had proposed a home camera-based fall detection system for the elderly based on artificial vision algorithm. Debard *et al.* [17] had developed a fall detection algorithm based on a simple background subtraction method. However, the algorithm only works properly when tested with publicly available datasets. Real-life data poses significant challenges and resulting in high false-positive fall detection. This issue is caused by the occlusions and changes in illumination conditions that interfere with the background segmentation. The robustness of the proposed algorithm is improved by using a particle filter and a person detector on foreground segmentation. The new approach obtained promising results on the newly created publicly available simulation dataset and real-time dataset with a sensitivity of 76.1% [18]. Ambient sensor-based fall detection is also known as environmental fall detection approach [19]. Yazar *et al.* [20] proposed a multi-sensor ambient assisted living system for fall detection by using Uno32 microprocessor. The proposed system is installed to an intelligent home to improve the safety of elderly and handicapped people. Litvak *et al.* [21] developed a fall detection system through acoustic sensing and floor vibration. The proposed algorithm shows a sensitivity and specificity of 95% but the system is not sensitive towards low impact falls.

3. RESEARCH METHOD

This section introduces the overall methodology implemented for the development of fall detection algorithm. The details are described briefly in the following subsections.

3.1. System block diagram

Figure 1 shows the block diagram of the proposed fall detection system. Acceleration and gyroscope data from MPU6050 are retrieved by using Arduino Uno through I2C communication protocol. These data will be converted to floating-point and transmitted to another Arduino Uno which is connected directly to the FPGA DE1-SoC board through wireless communication. FPGA will process the receive data and compare with the predetermined threshold value to identify a falling incident. If true positive falling accident is confirmed, FPGA will output a fall alert signal to indicate that immediate medical attention is required.

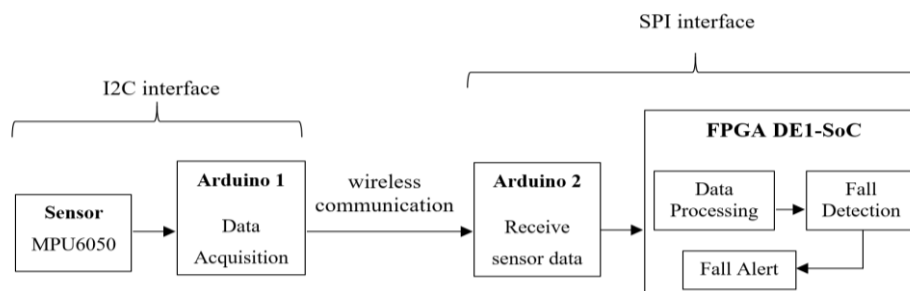


Figure 1. System block diagram of a fall detection system

The system will start by collecting the accelerometer and gyroscope data from MPU6050. If the reading of acceleration is less than the lower threshold value, the system will move to monitor state. The system expects to receive an acceleration reading which breaks the higher threshold value within 2s. If higher threshold is broken within 2s, fall accident is suspected. The system will proceed to check the orientation of human body based on the current gyroscope data. The type of falling accident will be determined by comparing the gyroscope data with the threshold value for different kinds of fall detection. The LEDs will light up to indicate that the alert system is activated and the timer will start to count down for nine seconds once the system suspects that a fall accident has happened. If the orientation of human body remains as falling when timer reached 0, the system will display a fall alert signal to indicate that true positive falling accident is confirmed. At this stage, the alert system can only be deactivated by caregivers or family members of user with the cancel switch.

3.2. Data acquisition

Accelerometer and gyroscope data are measured using MPU6050, an inertial measurement unit (IMU) sensor where both accelerometer and gyroscope are embedded into a single chip. MPU6050 consists of digital motion processor (DMP) to process complex 6-axis MotionFusion algorithms. It offloads the computation of motion-sensing algorithms from the host processor and improves the accuracy of output data by fusing the accelerometer and gyroscope data altogether. The full-scale range selected for accelerometer and gyroscope is ± 2 g and ± 250 degree/s respectively. The output acceleration, a_g from -1 g to +1 g is given in (1) [22].

$$a_g = a_{raw}/16384 \quad (1)$$

The total three-dimensional acceleration (a_x , a_y , a_z) measured from the accelerometer is the combination of human body acceleration and the earth gravity field, g in the unit of m/s^2 . The formula is given in (2) [22].

$$a_i = b + n + K[v(a_{body} + g)] \quad (2)$$

Where b denotes the bias vector, n denotes noise in measurement, K represents the scale factor matrix and v represents the direction cosine matrix of navigation frame orientation with respect to the body frame. The acceleration result is then compared to the sum vector magnitude, a_m given in (3). The average value obtained from (2) and (3) is taken as the final acceleration reading.

$$a_m = \sqrt{(a_x)^2 + (a_y)^2 + (a_z)^2} \quad (3)$$

DMP acquires the gyroscope data and computes the results in terms of quaternions. The computed results are stored in data register. On the other hand, a complementary filter is applied to fuse accelerometer and gyroscope data to further eliminate drift error in gyroscope. The gyro data in degree/s from (4) [22] is multiplied with the elapsed time value captured before each reading iteration to get the angle in degree as shown in (5) [23],

$$gyro = \frac{gyro_{raw}}{131} \quad (4)$$

$$angle = angle + gyro(elapsed\ time) \quad (5)$$

The attitude reading is computed with 96% of gyroscope data and 4% of accelerometer data as shown in (6) [23].

$$attitudereading = 0.96(angle_i) + 0.04(ag_i) \quad (6)$$

3.3. Threshold selection

The parametric method is used to determine the threshold value for the parameter lower and higher threshold in acceleration, time interval between lower and higher threshold, attitude reading (yaw, roll, pitch) for different type of fall events as shown in Table 1. The statistical population has an average weight of 45-80 kg and height of 150-180 cm. There are 20 persons (n samples) selected from the statistical population to estimate the optimum threshold value for the system.

Table 1. Set of features extracted for fall detection

Parameter		Equation
Lower threshold		Mean, $\overline{a_{min}} = \frac{\sum_{i=1}^n a_{i(min)}}{n}$
Higher threshold		Mean, $\overline{a_{min}} = \frac{\sum_{i=1}^n a_{i(min)}}{n}$
Time interval between lower and higher threshold		Average time interval, $t_i = t_{i(max)} - t_{i(min)}$ Mean, $\bar{t} = \frac{\sum_{i=1}^n t_i}{n}$
Attitude reading	yaw	Mean, $\overline{yaw} = \frac{\sum_{i=1}^n yaw_i}{n}$
	roll	Mean, $\overline{roll} = \frac{\sum_{i=1}^n roll_i}{n}$
	pitch	Mean, $\overline{pitch} = \frac{\sum_{i=1}^n pitch_i}{n}$

Each activity or movement will be repeated for three times and the average reading will be taken. The experimental data are analysed using Minitab software to determine the mean and standard deviation of the dataset for each of the required parameters. Tolerance intervals are used to define the upper and lower boundary of the threshold value for fall detection within a given confidence interval. The significance level chosen for this system is 0.05 and its corresponding confidence level is 95%. The threshold value for yaw, roll and pitch are determined in the same way as acceleration threshold. The range of gyroscope data for a particular fall event is taken from the time 500 ms after higher threshold acceleration is broken until a steady state is achieved. The gyroscope value at the time when higher threshold acceleration breaks is identified as the critical value at the stage where a person is prone to fall. This value is also defined as the lower boundary for non-fall event and upper boundary for fall event. Whereas the gyroscope value at the time where acceleration has achieved a steady state is defined as the critical value to identify the orientation of human body at lying state. The threshold value of these parameters is tabulated in Table 2.

The sensor was chosen to be placed on the arm of users to improve the accuracy of the threshold acceleration and gyroscope data. This may reduce the risk of false positive fall detection due to rapid body movement as arm is the human body part that experienced the least motion during body movement. Based on the result as shown in Table 2, the lower threshold and upper threshold of acceleration in fall accidents are 5 m/s² and 15 m/s² respectively. The average time interval between the lower and upper threshold is less than 300 ms. The change in gyroscope data was used to identify the type of fall accidents and differentiate fall accidents from NDA. In fall accidents, the threshold value of roll at steady state was less than 20° as human body is at lying state. The experimental results have proved that the combination of accelerometer and gyroscope data could provide better reflection on human body movement and posture. This could reduce the rate of false positive fall detection due to fall-like activities that resulted the same trend as fall accident in acceleration data.

Table 2. Threshold value of parameters in fall detection

Parameter	Acceleration (m/s ²)		Average time interval between lower and upper threshold (ms)	Attitude reading (deg)		
	Lower threshold	Upper threshold		Roll	Pitch	Yaw
Fall events						
Forward	< 5.00	> 15.00	< 300	< 20	> 60	< -60
Backward	< 5.00	> 15.00	< 300	< 20	< -60	> 60
Right Side	< 5.00	> 15.00	< 300	< 20	> 0	> 0
Left Side	< 5.00	> 15.00	< 300	< 20	< 0	< 0

4. RESULT AND DISCUSSION

4.1. Fall detection analysis

Figure 2 shows the change in acceleration of a fall accident from a static posture. Fall detection analysis can be separated into three phases: start, fall and impact. Phase 1 was the early stage in a fall detection, where a person begins or tends to fall to the ground. The acceleration of the human body started to decline and reached its minimum value which was lower than the lower threshold (less than 5 m/s²) at phase 2. Human body will experience a fall impact when it reached the ground. The acceleration started to increase at the mid of phase 2 and reached its maximum value which was higher than the upper threshold (higher than 15 m/s²) at phase 3. There was a great change in acceleration within a short period (less than 300 ms) in phase 2 and phase 3. The acceleration started to decrease after it reached its maximum magnitude. At phase 3, the acceleration fluctuated within the upper and lower boundary due to the decrease in fall impact after the first hit on the ground and remained within $\pm 2\%$ of its final value when the body reached its steady state.

Based on the experimental result as shown in Figure 3 (a), fall-like activities such as fast walking, running and jumping may also result in rapid change in the magnitude of acceleration like fall accidents. This may lead to false positive fall detection if only acceleration data was being considered. In order to differentiate fall accidents and fall-like activities, gyroscope data was taken into consideration at the same time to determine human posture. The combination of accelerometer and gyroscope data could minimize the rate of false positive fall detection on fall-like activities in acceleration mean effectively. A person who experienced an accidental fall normally will end up at lying state with gyroscope data, roll (x-axis) less than 20°. For non-fall events, the roll will always above 50° as shown in Figure 3 (b). Based on the results obtained, all the falling accidents were showing the same trend in the change of acceleration magnitude. Although the time interval between the minimum acceleration at 'Fall phase' and the maximum acceleration at 'Impact phase' was slightly different from each other, it was remained within the threshold value (less than 300 ms). Besides, fall-like activities may also result in a great change in the magnitude of acceleration, which met the threshold of a fall accident. In this case, the reading of roll that define the position of human body in x-axis was used to distinguish fall accidents and fall-like activities.

Although fall accidents were having the same trend in the change of acceleration magnitude, the type of falling accidents can be differentiated based on the change in pitch and yaw data. During the fall accident, the reading of roll started to decline as the human position started to change from standing to lying as shown in Figure 4 (a). In forward falling, pitch started to increase to its maximum value, while yaw started to decrease to its minimum value before reached its steady state. The reading of roll for a fall accident will be less than 20° after steady state was achieved as a person who experienced fall usually will ended with the lying position. All the attitude reading achieved its steady state value (remained within $\pm 2\%$ of its final value) during the 'Impact phase'.

In backward falling, it was observed that the reading of yaw started to increase while the reading of pitch started to decrease during the 'Fall phase' as shown in Figure 4 (b). The attitude reading is slightly fluctuated before it came to steady state. For side falling, both pitch and yaw started to decrease during the Fall phase as shown in Figure 4 (c) and Figure 4 (d). Leftward or rightward falling can be distinguished through the final value of pitch and yaw. Leftward falling shows negative final value, while rightward falling shows positive final value. The result for fall detection analysis was tabulated in Table 3 to compare with its threshold value.

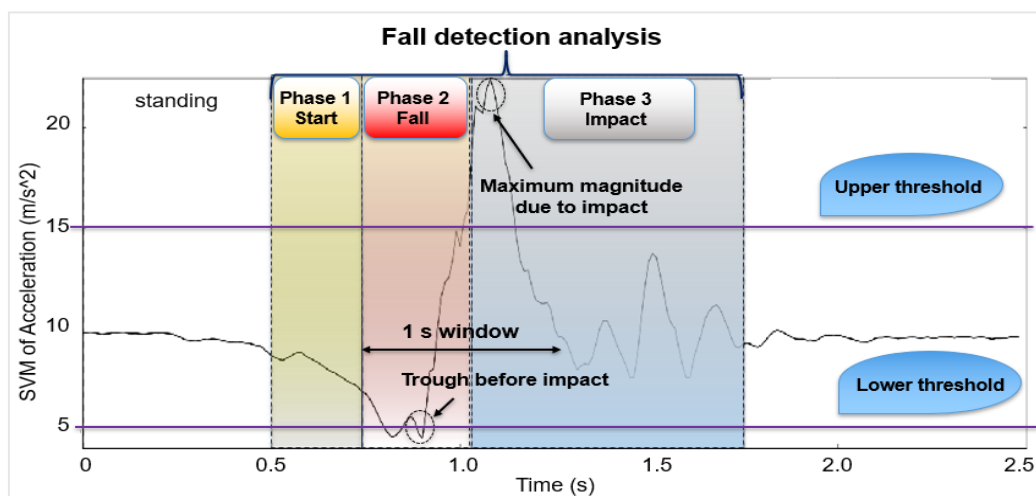


Figure 2. Fall detection analysis

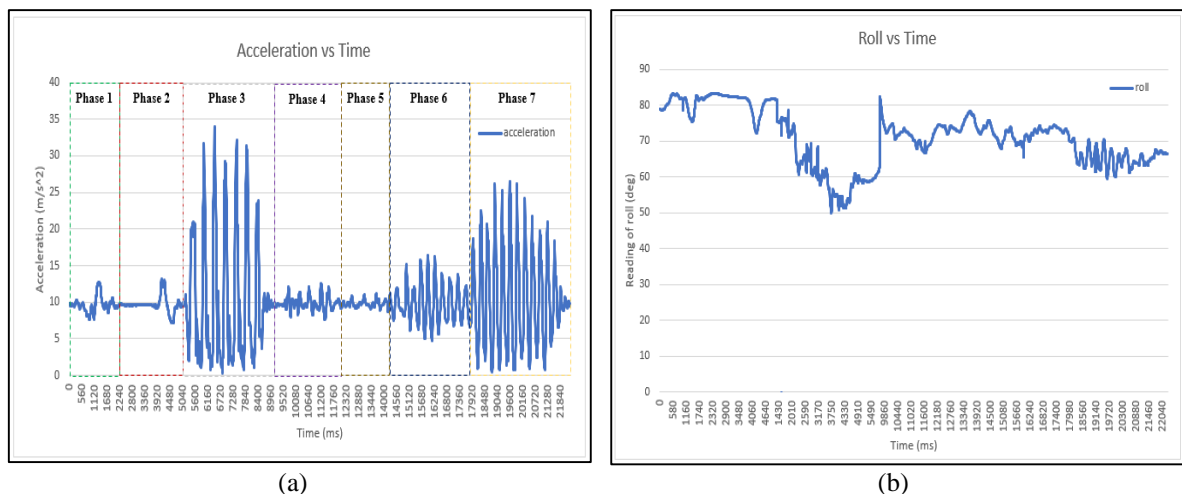


Figure 3. Change in, (a) acceleration for NDA, (b) roll for NDA

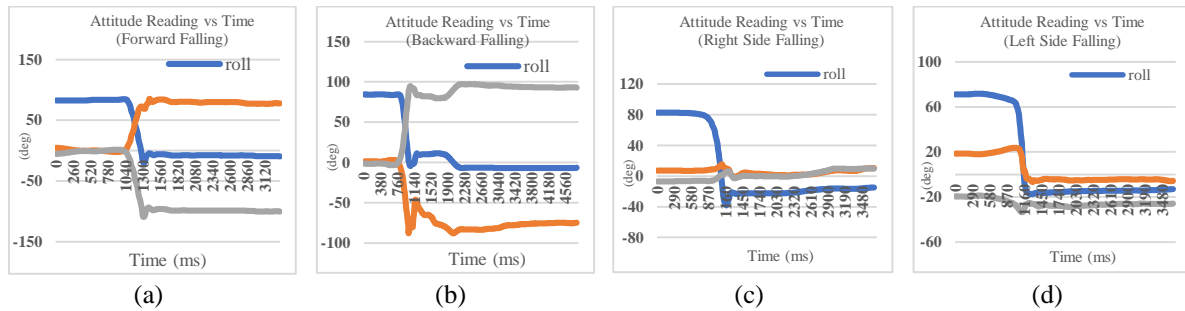


Figure 4. Change in gyroscope data, (a) forward falling, (b) backward falling, (c) right side falling, (d) left side falling

Table 3. Result of fall detection analysis

Fall events	Acceleration (m/s^2)		Average time interval between lower and upper threshold (ms)	Attitude reading (deg)		
	Minimum	Maximum		Roll	Pitch	Yaw
Forward	3.18	33.61	90	-6.73	81.94	-95.88
Backward	3.54	33.78	110	-6.65	-83.13	97.09
Right Side	2.11	28.59	70	-22.5	3.52	0.05
Left Side	1.98	23.58	80	-16.34	-3.9	-28.17

4.2. FPGA implementation

The proposed algorithm was implemented on FPGA DE1-SoC board using Altera Quartus II design software to evaluate its performance in fall detection. There were four interfaces module in this design. Since the frequency used in FPGA DE1-SoC board is 50 kHz, a clock module was required to slow down the clock and generate clock pulse with period of 1 s for the design. The 'mpu6050_data' module was responsible to receive and process the input sensor data. It will generate the relevant internal signal for state transition in 'FSM_ctrl' module. The timer module acted as a countdown timer to control the state transition in 'FSM_ctrl' module. The type of falling events includes forward falling, backward falling and side falling. The LEDs and 7-segments on the FPGA board that were used to represent the outputs. The change in sensor data for forward, backward and side falling is shown in Table 4.

Table 4. Change in sensor data for forward falling

Falling	Time (ms)	Acceleration (m/s^2)	Attitude reading (degree)			Description
			Roll	Pitch	Yaw	
Forward	1190	4.02	39.19	50.09	-51.00	Lower threshold is violated
	1320	33.61	-7.7	72.68	-96.04	Higher threshold is violated
	1510	10.36	-7.7	80.27	-98.23	Steady state
Backward	900	3.94	52.13	-37.16	37.92	Lower threshold is violated
	1150	32.15	11.81	-78.80	90.97	Higher threshold is violated
	1290	10.43	1.22	-81.47	96.93	Steady state
Side Left	1080	4.07	42.11	20.92	-32.72	Lower threshold is violated
	1230	23.11	-10.93	-1.40	-27.69	Higher threshold is violated
	1410	10.38	-17.55	-3.89	-29.05	Steady state
Side Right	1050	3.65	24.77	12.09	-0.61	Lower threshold is violated
	1170	28.79	-12.67	11.67	2.49	Higher threshold is violated
	1390	10.49	-21.46	4.83	3.24	Steady state

The acceleration of human body reached its minimum value and violated the lower threshold at time 1190 ms. 'timer1' started to count down and the LED that represented 'alert1' was turned on. The number displayed on the second 7-segment (from right) changed to '1' to indicate that lower threshold was violated. At 1320 ms, the acceleration reached its maximum value and violated the higher threshold. The reading of roll, pitch and yaw met the criteria of forward falling. The right most 7-segment displayed a number '1' to show that forward falling was detected. 'timer10' started to count down and the LED that represented 'alert2' was turned on. The number displayed on the second 7-segment (from right) changed to '2' to indicate that upper threshold was violated and forward falling was suspected. The alert system was activated, and user was given 10 s to change their posture. If human posture was detected at lying state (roll < 20°) when 'timer10' reached 0, forward fall accident is confirmed. The LED that represented "fall" turned on and the number

displayed on the second 7-segment (from right) changed to '3' to indicate forward falling had happened. On the other hand, the system will back to normal state if human posture was detected at standing or sitting position before 'timer10' was time out. The number displayed on the second 7-segment (from right) changed to '4' to indicate that no fall accident was detected.

For backward falling, at time 900 ms, the acceleration of human body reached its minimum value and violated the lower threshold. The 'timer1' started to count down and the LED that represented 'alert1' was turned on. The number displayed on the second 7-segment (from right) changed to '1' to indicate that lower threshold was violated. The acceleration reached its maximum value and violated the higher threshold at time 1150 ms and current reading of roll, yaw and pitch were met the criteria of backward falling. The right most 7-segment displayed a number '2' to show that backward falling was detected. 'timer10' was started to count down and the LED that represents 'alert2' was turned on. The number displayed on the second 7-segment (from right) changed to '2' to indicate that upper threshold was violated and backward falling was suspected. Backward fall accident was confirmed if human posture was detected at lying state when 'timer10' was timeout. The LED that represented 'fall' turned on and the number displayed on the second 7-segment (from right) changed to '3' to indicate backward falling has happened. In left side falling, the acceleration of human body reached its minimum value and breaks the lower threshold at time 1080 ms. 'timer1' started to count down. The LED that represents 'alert1' is turned on and the number displayed on the second 7-segment (from right) was changed to '1' to indicate that lower threshold was violated. The acceleration had reached its maximum value and broke the higher threshold at time 1230 ms. The change in attitude reading met the criteria of left side falling. 'timer10' started to count down and the number displayed on the second 7-segment (from right) changed to '2' to indicate that upper threshold is violated. The LED that represented 'alert2' was turned and side falling was suspected. When 'timer10' reached 0, the LED that represented 'fall' turned on if human posture was detected at lying state. The number displayed on the second 7-segment (from right) changed to '3' to indicate that side falling had happened. The same concept was applied to right side falling.

4.3. Performance evaluation

The performance of the fall detection algorithm was evaluated in terms of specificity and sensitivity through FPGA implementation. Since fall detection is a case of binary classification, each motion can be classified as either a fall event (true positive) or non-fall event (true negative). There were four possible outcomes in fall detection, true positive (TP), fall event is detected correctly, false positive (FP), non-fall event is detected as fall, true negative (TN), non-fall event is detected correctly, false negative (FN), fall event is not detected.

Sensitivity is the rate of true positive fall detection, while specificity represents the rate of true negative fall detection. The sensitivity and specificity of the algorithm were calculated by using (7) and (8) [24]. The rate of false alarm due to non-fall events mistakenly detected as falls can be computed using (9) [24]. The miss rate of detection in non-fall event is also known as false positive rate.

$$\text{sensitivity} = \frac{\text{true positive(TP)}}{\text{true positive(TP)} + \text{false negative(FN)}} \quad (7)$$

$$\text{specificity} = \frac{\text{true negative(TN)}}{\text{true negative(TN)} + \text{false positive(FP)}} \quad (8)$$

$$\text{Miss rate}_{\text{non fall event}} = \frac{\text{false positive(FP)}}{\text{false positive(FP)} + \text{true negative(TN)}} \quad (9)$$

Sensitivity is the rate of true positive fall detection that represent the proportion of fall event, which are correctly detected by the algorithm. It was measured from its capability in detecting fall events such as forward falling, backward falling and side falling. By referring to the performance analysis in Figure 5 (a), the proposed algorithm managed to detect all types of fall accidents accurately, except for falls that ended with sitting position. When a person falls and end up with a sitting position, the algorithm will assume that it was a non-fall event as the criteria of fall accident was not met in gyroscope mean. In this case, false negative fall detection can only be avoided if the change in magnitude of acceleration does not met the criteria of fall accident. Specificity is denoted as the rate of true negative fall detection. A set of daily movement or activities which includes sitting, standing, walking, running, lying and fall-like activities such as fast walking, fast running, jumping, fast sitting, fast standing and fast lying was carried out with 20 samples to determine the specificity of the system in detecting non-fall events. On the other hand, the proposed algorithm achieved an average accuracy of 100% in true negative fall detection except for fast lying, 75%. This is because fast lying involved a rapid change in the magnitude of acceleration and end up with lying

state. Thus, it may lead to false-negative fall detection when the change in the magnitude of acceleration and gyroscope data met the criteria of a fall event. The overall performance in detecting non-fall events was summarized in Figure 5 (b).

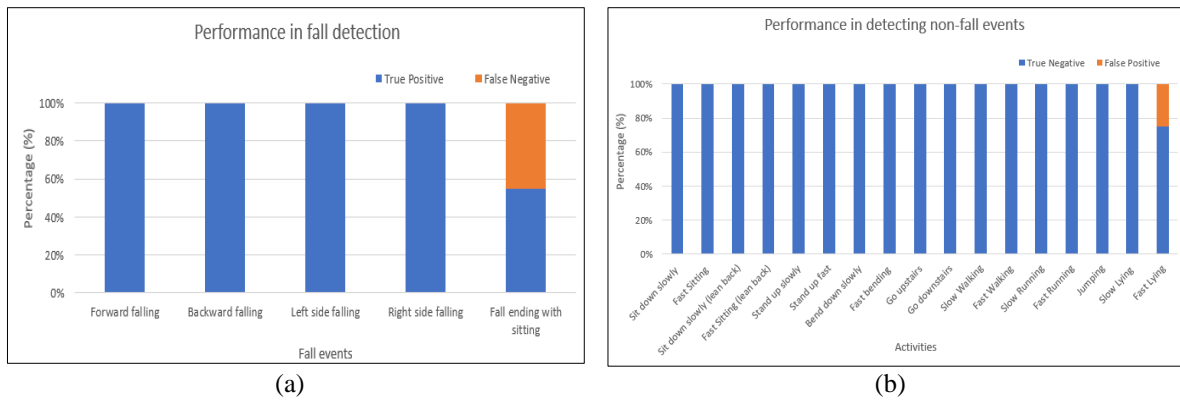


Figure 5. Performance in; (a) fall events detection, (b) non-fall events detection

The sensitivity and specificity of the algorithm were calculated by using confusion matrix as shown in Table 5. From Table 5, there was a total of 200 actual fall events and 340 actual non-fall events being recorded. The predicted event is referring to the number of fall events or non-fall events that were successfully detected by the proposed algorithm. From the 340 recorded actual non-fall events, 335 cases were detected correctly as non-fall events with 5 miss cases. The miss rate in true negative fall detection was contributed by the failure of the proposed algorithm in recognizing fast lying as non-fall event due to its end position that was similar to fall accident. On the other hand, there were 9 miss cases in true positive fall detection. By using the formula shown in (7) and (8), the sensitivity and specificity of the algorithm in fall detection were 97.45% and 97.38% respectively. This means that 191 of the fall accidents were detected correctly out of the 200 recorded actual fall events. The proportion of false negative fall detection was contributed by the miss rate in detecting fall that ended with sitting. Thus, it can be concluded that the proposed algorithm had achieved the desired sensitivity with an accuracy of 100% in detecting forward, backward and side falling. Fall that ended with sitting position is an additional test to evaluate the sensitivity of the proposed algorithm as compared to Li *et al.* [25].

Table 5. Confusion matrix for performance evaluation

Predicted \ Actual	Fall event	Non-fall event	Actual Number
	Fall event	Non-fall event	
Fall event	191 (TP)	9 (FP)	200
Non-fall event	5 (FN)	335 (TN)	340

The previous research only considered the sum vector of aggregate rotational angle instead of axis-based gyroscope data analysis in fall detection. In this work, proposed offered better analytical results in terms of human inclination by including non-fall like activities in the analysis. Based on the threshold performance, it shows a good result with the sensitivity and specificity of the algorithm were 97.45% and 97.38% respectively. The miss rate in sensitivity was mainly contributed from falls that ended with sitting. This test was only conducted by Li *et al.* [25] and the sensitivity achieved from 72 records was 92%, while the specificity achieved was 91% from 70 records. From the performance evaluation in detecting fall events, the proposed algorithm has achieved 100% in true positive fall detection for all type of fall events, including forward, backward, leftward and rightward falling. The specificity of the proposed algorithm was slightly lower compared to Torres *et al.* [15] due to no number of test coverage on non-fall like activities. Vetsandonphong [13], shows the sensitivity is 95% and specificity 90% based on 20 activities of ADL and falling down motion. In conclusion, the proposed algorithm shows good threshold values by included both fall event and non-fall like activities in the threshold performance.

5. CONCLUSION

A threshold-based fall detection algorithm was successfully developed in FPGA to evaluate the performance of sensitivity and specificity. Based on the result obtained, the proposed algorithm has achieved a sensitivity of 97.45% and specificity of 97.38%. The proposed algorithm managed to detect fall events and non-fall events correctly, except for fast lying and falling event ending with sitting. This proved that the combination of accelerometer and gyroscope data has provided a better reflection on human posture in determining different type of fall accidents, while minimizing the rate of false positive fall detection due to fall-like activities such as jumping, running and fast walking. The combination of accelerometer and gyroscope data had reduced the rate of false positive fall detection. The previous research only considered the sum vector of aggregate rotational angle instead of axis-based gyroscope data analysis in fall detection. Thus, proposed algorithm offered better analytical results in terms of human inclination. The accuracy of the proposed algorithm can be further improved by combining the thresholding analytical method with a machine learning method in determining the threshold value. Besides, an additional gyroscope can be used to detect the posture of the lower body part. This could eliminate the limitation of the proposed algorithm in detecting falls that ended with sitting.

ACKNOWLEDGEMENTS

Special thanks to CEDEC USM EDA on providing the Synopsys software access and School of Electrical and Electronic Engineering USM on providing FPGA DE1-SoC during the research development.

REFERENCES

- [1] Bernama.com, "Malaysia's estimated population in 2019-32.6 million," 2019, [Online] Available: <http://www.bernama.com/en/news.php?id=1746406>.
- [2] World Health Organization, "Ageing and Health," 2019, [Online] Available: <https://www.who.int/news-room/fact-sheets/detail/ageing-and-health>.
- [3] World Health Organization, "Falls," 2019, [Online] Available: <https://www.who.int/news-room/fact-sheets/detail/falls>.
- [4] S. S. Khan, and J. Hoey, "Review of Fall Detection Techniques: A Data Availability Perspective," *Medical Engineering & Physics*, vol. 39, pp. 12-22, doi: 10.1016/j.medengphy.2016.10.014.
- [5] C. W. Chen, and Y. C. Chang, "Support Vector Regression and Genetic Algorithm for HVAC Optimal Operation," *Mathematical Problems in Engineering*, vol. 2016, pp. 315-320, doi: /10.1155/2016/6212951.
- [6] S. Fudickar, A. Lindemann, and B. Schnor, "Threshold-based Fall Detection on Smart Phones," *Proceedings of the International Joint Conference on Biomedical Engineering Systems and Technologies*, vol. 5 (BIOSTEC 2014). SCITEPRESS-Science and Technology Publications, Lda, Setubal, PRT, 2014, 303-309, doi: 10.5220/0004795803030309.
- [7] P. Vallabh, R. Malekian, N. Ye, and D. C. Bogatinoska, "Fall detection using machine learning algorithms," *24th International Conference on Software, Telecommunications and Computer Networks SoftCOM*, 2016, pp. 1-9, doi: 10.1109/SOFTCOM.2016.7772142.
- [8] Y. S. Delahoz, and M. A. Labrador, "Survey on Fall Detection and Fall Prevention Using Wearable and External Sensors," *Journal of Sensors*, vol. 14, no. 19808, pp. 1-37, 2014, doi: 10.3390/s141019806.
- [9] S. Irene, N. M. Shwetha, P. Haribabu, and R. Pitchiah, "Development of ZigBee Triaxial Accelerometer Based Human Activity Recognition System," *IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 2015, pp. 1460-1466, doi: 10.1109/CIT/IUCC/DASC/PICOM.2015.357.
- [10] S. Abdelhedi, R. Bourguiba, J. Mouine, and M. Baklouti, "Development of a two-threshold-based fall detection algorithm for elderly health monitoring," *IEEE Tenth International Conference on Research Challenges in Information Science RCIS*, 2016, pp. 1-5, doi: 10.1109/RCIS.2016.7549315.
- [11] T. T. Dang, H. Truong, and T. K. Dang, "Automatic Fall Detection using Smartphone Acceleration Sensor," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 12, pp. 123-129, 2016.
- [12] Q. T. Huynh, U. D. Nguyen, L. B. Irazabal, N. Ghassemian, and B. Q. Tran, "Optimization of an Accelerometer and Gyroscope-Based Fall Detection Algorithm," *Journal of Sensors*, vol. 2015, Article ID 452078, pp. 1-9, 2015, doi: 10.1155/2015/452078.
- [13] N. Vetsandonthong, "Arduino Based Fall Detection and Alert System," Available from Universiti Teknologi PETRONAS Theses and Dissertations, pp. 1-52, 2016.
- [14] D. Yacchirema, and J. S. Puga, "Fall detection system for elderly people using IoT and Big Data," *Procedia Computer Science*, vol. 130, pp. 603-610, 2018, doi: 10.1016/j.procs.2018.04.110.
- [15] G. G. Torres, R. V. B. Henriques, C. E. Pereira, and I. Muller, "An EnOcean Wearable Device with Fall Detection Algorithm Integrated with a Smart Home System," *IFAC PapersOnLine*, vol. 51, no. 10, pp. 9-14, 2018, doi: 10.1016/j.ifacol.2018.06.228.
- [16] K. D. Miguel, A. Brunete, M. Hernando, and E. Gambao, "Home Camera-Based Fall Detection System for the Elderly," *Sensors*, vol. 17, no. 2864, pp. 1-21, 2017, doi: 10.3390/s17122864.

- [17] G. Debard, M. Mertens, T. Goedeme, T. Tuytelaars, and B. Vanrumste, "Three Ways to Improve the Performance of Real-Life Camera-Based Fall Detection Systems," *Journal of Sensors*, vol. 2017, Article ID 8241910, pp. 1-16, 2017, doi: 10.1155/2017/8241910.
- [18] G. Debard, G. Baldewijns, T. Goedemé, T. Tuytelaars, and B. Vanrumste, "Camera-based fall detection using a particle filter," *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBC*, 2015, pp. 6947-6950, doi: 10.1109/EMBC.2015.7319990.
- [19] P. Vallabh, and R. Malekian, "Fall detection monitoring systems: a comprehensive review," *Journal of Ambient Intelligence and Humanized Computing* volume, vol. 9, pp. 1809-1833, 2018.
- [20] A. Yazar, F. Erden, and A. E. Cetin, "Multi-sensor ambient assisted living system for fall detection," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*; Florence, Italy 4-9 Mei, 2014, pp. 1-3.
- [21] D. Litvak, I. Gannot, and Y. Zigel, "Detection of falls at home using floor vibrations and sound," *IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, 2008, pp. 514-518, doi: 10.1109/EEEI.2008.4736581.
- [22] InvenSense Inc, "MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking™ Devices," 2020.
- [23] Dejan, "Arduino and MPU6050 Accelerometer and Gyroscope Tutorial," 2019.
- [24] R. W. Broadley, J. Klenk, S. B. Thies, L. P. J. Keney, and M. H. Granat, "Methods for the Real-World Evaluation of Fall Detection Technology: A Scoping Review," *Sensors*, vol. 18, no. 7, pp. 1-28, doi: 10.3390/s18072060.
- [25] Q. Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach, and G. Zhou, "Accurate, Fast Fall Detection Using Gyroscopes and Accelerometer-Derived Posture Information," *Sixth International Workshop on Wearable and Implantable Body Sensor Networks*, 2009, pp. 138-143, doi: 10.1109/BSN.2009.46.

BIOGRAPHIES OF AUTHORS



Lo Pui Mun received Bachelor' degree in Electrical and Electronic Engineering from Universiti Sains Malaysia, Malaysia in 2020. Currently, she is Graduate Trainee in Intel Corporation, Penang, Malaysia. Her research interest is in Digital IC design.



Azniza Abd Aziz received the Ph.D. degree in Electrical Engineering from the University of South Carolina, Columbia, SC, USA. She was an Advanced Signal Integrity Engineer with Intel Corporation, Penang, Malaysia, and a Senior Signal Integrity Engineer with Hewlett Packard Enterprise, Palo Alto, CA, USA, with ten years of experience in designing and validating desktop, mobile, and server platforms. She is currently with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Pulau Pinang, Malaysia. Her current research interests include signal integrity solutions for high-speed digital design, electromagnetic, communication systems and RF and microwave engineering. Dr. Azniza is a senior member IEEE and Chartered Engineer (CEng) conferred by the Engineering Council, U.K.