Development of a Fall Detection System Based on Neural Network Featuring IoT-Technology

Y. J. Ng[1,2] , N. S. N. Anwar[1*] , W. Y. Ng[1] , C. Q. Law[1]

[1]Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
[2]Intel Microelectronics (M) Sdn Bhd, Plot 6, Bayan Lepas Technoplex, 11900 Bayan Lepas, Pulau Pinang

**syahrim@utem.edu.my*

Abstract— **Accidental falls are considered a major cause of accidents that could lead to serious injuries, paralysis, psychological damage, and even deaths, especially for the elderly. Therefore in this project, a neural network-based fall detection system that could automatically detect a fall event is proposed. The system is enhanced with Internet-of-Things (IoT) features that could reduce the response time and efficiently improve the prognosis of fall victims. A 10 Degree of Freedom (DOF) Inertial Measurement Unit (IMU) module is connected to an Intel Edison with Mini Breakout board and mounted on a wearable waist-worn device to continuously record body movements. A backpropagation neural network algorithm has been developed to accurately distinguish falls from different postural transitions during activities of daily living (ADL). A body temperature and heart-pulse monitoring device were developed for this system to provide the medical personnel additional information on the body condition of the fall victim. Using the latest IoT-technology, the system can be connected to the internet and provides a continuous and real-time monitoring capability. Once a fall accident happens, the system will be automatically triggered. This will activate an Android App through the Wi-Fi network that will then send an emergency SMS with the actual location and body conditions of the victim to a recipient. A series of falls and ADL simulations were performed by a group of subjects to test and validate the performance of the system. The experiment results showed that the proposed system could obtain a sensitivity of 95.5%, specificity of 96.4%, and accuracy of 96.3%.**

Keywords— Fall detection, neural network, voice response, vital-sign monitoring, Internet-of-Things (IoT).

I. INTRODUCTION

FALL accident is one of the health risks which frequently happens, especially for the elderly who over the age of 65 years old. There are about 33% of the elderly are reported to experience fall injuries at least once per year and 68% hospitalization of the elderly is fall-related [1]. Moreover, there are some sports activities like hiking that will cause serious fall injuries for people. Every year, fall accidents will cause about 10,000 deaths among humans aged 65 years and above in the United States [1]. In Malaysia, it is expected that in 2035 the senior citizen will make up 15% of its population [2]. This trend indicates a need for reform in the healthcare monitoring and delivery system. There are many reasons for fall accidents, such as heart attack and heat stroke. These reasons are usually unknown to the doctors or the rescue team until they reach the accident scene. Therefore, in this paper, the study and design of an accurate and real-time human fall detection system with a vital sign monitoring function

Article history: Manuscript received 28 August 2020; received in revised form 30 September 2020; Accepted 30 September 2020.

will be presented. The system could reduce the response time to fall-related accidents and help the medical team in the decision-making process.

Over the last decade, the approach on fall detection and alert system is normally categorized into ambient-based, vision-based, and wearable device-based [3-6]. However, ambient-based and visual-based methods are impractical and can be significantly affected by the external environment [7, 8]. The interference factors which exist in the visual-based method will increase the difficulty for fall detection and the recognition rate is just about 50% -70% [3]. The Wearable sensors-based approach works by measuring and processing the inertial signal of human motion, such as the acceleration and the angular rate [9-12]. However, there is some difficulty distinguishing between falls and non-fall activities, such as running and lying down. Some researchers proposed a method, which predicts falls using a threshold of the angle and time [13-17]. Since the fall events could occur randomly, this method is unstable and has a low recognition rate. By applying the neural network algorithm, the presence of falls can be accurately detected even before the collision happens [18-21].

Meanwhile, the latest trend of connecting physical devices to the internet or Internet-of-Things (IoT) leads to improvements in healthcare monitoring [22-24]. Easy access to Wi-Fi connections in the cities and even rural areas makes it the medium of choice for IoT applications. In 2014, Bai et al. [4] introduced a human fall detection system with a voice response function. A connected device allows not only a real-time alert system but also the transfer of additional information such as the vital sign signals, locations of accidents, and nearest hospitals.

In the first part of this paper, the methodology used in the project is described. This includes the system overview, hardware, and the development of the android application. Next, the experiments and the measured human motion signals are presented. The final part of the methodology section describes the proposed fall detection algorithm using a neural network. Finally, the performance of the system is discussed in the result section.

II. METHODOLOGY

A. System Overview

A GY-80 10 DOF IMU module is mounted to the Intel Edison with Mini Breakout Board and worn on the waist of the human body to continuously record body movements and detect body postures. The module must be rigidly attached to the waist of the target to eliminate errors in the measurement. The vital-sign monitoring device which consists of an LM35 temperature sensor and pulse rate sensor will be worn on the wrist of the user to measure the body temperature and pulse rate. The vital-sign monitoring device will send the information on the body condition of the user to the smartphone when fall activities are detected. Once a fall accident happens, the alert system will trigger and send emergency messages, body conditions, and the actual location of the user to his parents or friends. Fig. 1 shows the block diagram of the system.

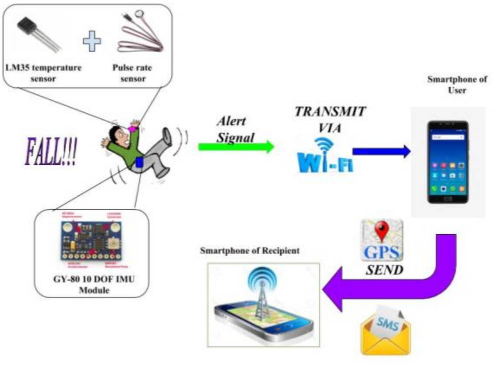


Fig. 1. Block diagram of the human fall detection system

B. Hardware Description

The 10-DOF IMU module with model-number GY80-10 is mounted to the Intel® Edison with Mini Breakout Board to be used for data acquisition. It consists of an L3G4200D (3-Axis Gyroscope), ADXL345 (3-Axis

Digital Accelerometer), HMC5883L (3-Axis Magnetometer), and BMP085 (Barometer). The IMU module is shown in Fig. 2(a). It has a dimension of (25.8 x 16.8) mm and a 3 mm installation hole. In this system, only the accelerometer and angular rate sensor are used to collect falling data. Although most smartphones consist of integrated accelerometer and gyroscope, there are some difficulties for the real-time monitoring of the activities of the elderly using those sensors. Moreover, further, improvement using more sensors in the system is possible if the IMU module is used. Therefore, external sensors might more appropriately be applied in the system.

The system is powered by a rechargeable lithium-polymer battery for a lightweight and longer operational time for the system. The Intel Edison board is packed with a huge amount of tech features in a small size while still delivering the same robust strength as your go-to single board computer. The Edison board is powered by an Intel® Atom™ SoC dual-core CPU and consists of an integrated Wi-Fi, Bluetooth LE, and a 70-pin connector to attach a veritable slew of shield-like "Blocks" which can be stacked on top of each other [9]. Its low power and small footprint make it ideal for projects that need a lot of processing power, but with a minimum power source and small footprint. The Intel® Edison with Mini Breakout Board is shown in Fig. 2(b).

For the vital sign monitoring device included in this system, an LM35 temperature sensor and pulse rate sensor module are connected to NodeMcu Lua ESP-12E ESP8266 Wi-Fi Development Board Ver2 to measure the body temperature and pulse rate of the subjects. NodeMCU is an open-source platform for the Internet of Things (IoT). It consists of ESP-12 module-based hardware and firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems. The term "NodeMCU" by default refers to the firmware rather than the development kits. The programming of firmware is based on the Lua scripting language. However, NodeMCU also able to be programmed using the Arduino IDE.

(a) (b)

Fig. 2. (a) GY80-10 DOF Inertial measurement unit module and (b) Intel® Edison with Mini Breakout Board

The module must be rigidly attached to the waist of the target to eliminate errors in the measurement. The sole reason for the best performance of the sensor unit on the waist position may be due to the waist is the centre of gravity of the human body and truly reflects the posture of the trunk.

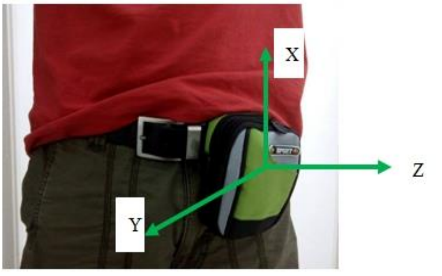


Fig. 3. Placement of the IMU and processor board

The LM35 temperature sensor is utilized as a part of the system to quantify the body temperature of the human body. The LM35 temperature sensor is a precision integrated-circuit temperature sensor, which has an operating temperature over a -55° to $+150^{\circ}$C temperature range. The output voltage of LM35 is directly proportional to the Celsius (Centigrade) temperature. A Pulse sensor is utilized in the system to measure the heart rate of the user. For the pulse rate measurement, the sensor clips onto a fingertip or earlobe and connects to the microcontroller with some jumper cables.

For the vital-sign monitoring device, the prototype was completed using a small container and a wristband. An LM35 temperature sensor and a pulse rate sensor were connected to the NodeMCU board and placed inside the small container. Fig. 4 shows the prototype of the vital-sign monitoring device.

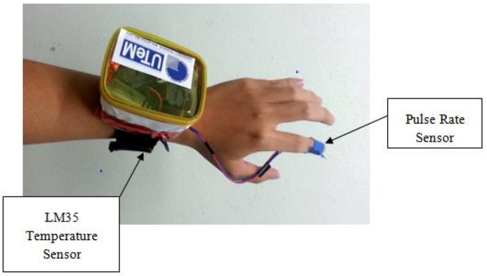


Fig. 4. Prototype of the vital-sign monitoring device

C. Development of Android Application

The system is implemented on the Android platform that allows receiving via a Wi-Fi interface. An Android Application was developed using the Android Studio software. The Android Application will be connected to the microcontroller through TCP/IP (Transmission Control Protocol/Internet Protocol) using the HTTP method. The Android Application consists of the shortest message service (SMS) and global positioning system (GPS) location service of the smartphone. The actual location is determined using the integrated GPS module of the smartphone. The data processing process will be implemented in the format of service, which will run in the background. This system includes the voice response function that consists of a text-to-speech (TTS) system. The text-to-speech (TTS) system analyses language text and convert the language text into speech. This system will acquire the user's input before sending any emergency notification. Fig. 5 shows the block diagram of a typical TTS system.

The alert screen will pop out when the system detects any fall occurs as shown in Fig. 6(a). The users may press the "Yes" or "No" button depending on the necessity of sending an alert message. In addition, the system also includes a 20-second countdown timer function. If the user already loses consciousness because of the fall and the countdown timer has reach zero without any response, the application will automatically activate the alert function and send the alert message together with GPS location and body conditions of the users to the third party. Fig. 6(b) presents the alert messages sent to the third party.

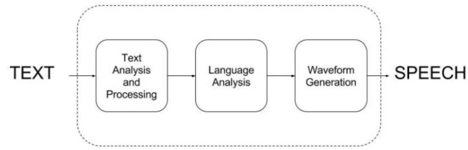
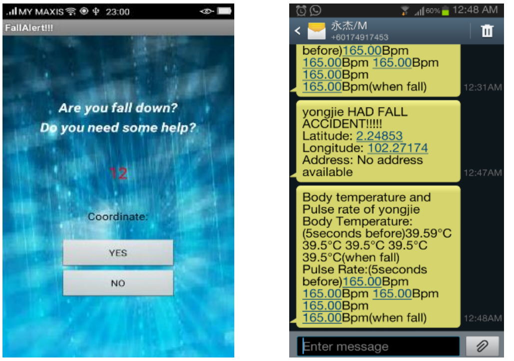


Fig. 5. Typical TTS system



(a) (b)

Fig. 6. (a) The fall-alert window with a countdown timer and (b) examples of the alert messages sent to the relevant parties

D. Fall and ADL data collection.

To develop the fall detection algorithm and validate the proposed solution, two simulation experiments were carried out with the wearable electronic module placed on each subject's waist as shown previously in Fig. 3. The experiments were conducted by fifteen human subjects, who performed various ADL motions such as walking, sitting, running, and jumping. In the second experiment, different types of fall motions such as fall forward, fall backwards, fall left, and fall right had been performed.

Acceleration and gyroscope data (in three dimensions) were gathered from the accelerometer and gyroscope of the GY80-IMU module. The collected data are sampled at 100 Hz and recorded in a Micro SD card for further processing. Fig. 7 shows the simulation of the fall as performed by one of the subjects.

1.


2.

3.

Fig. 7. Simulations of fall

E. Signal Processing

Signal processing of the signals from the GY80-IMU module such as transformation, filtering, and visualization will be performed using the Signal Processing Toolbox of MATLAB software. The Signal Processing Toolbox can be used for signal analysis in time, frequency, and time-frequency domains, patterns, and trend identification, feature extraction, and custom algorithm development and validation.

The signal data were filtered to reduce noise and compensate for gyroscope drift. In the latter case, a high-pass filter was applied to the acceleration and angular velocity data from the IMU module. The data are high-pass filtered with a second-order Butterworth filter. The gyroscope data are prone to drift and the angle derived from angular velocity continues to change during integration even when the sensor is stationary.

During fall detection, the information on the acceleration or rotation of the body in any specific direction is less important. Rather, an aggregate that combines the 3-dimensional sensor outputs may be more suitable to be considered for the analysis. For this reason, vector magnitude is used as a data feature for feature extraction; the vector magnitude of acceleration and angular velocity can be calculated with the formula below:

$$A=\sqrt{a_x{}^2 + a_y{}^2 + a_z{}^2} \tag{1}$$

$$G = \sqrt{g_x{}^2 + g_y{}^2 + g_z{}^2} \tag{2}$$

where a_x ,a_y and a_z are the accelerometer readings for each axis and g_x,g_y and g_z are the gyroscope readings for each axis

F. Artificial Neural Network

A neural network was designed to detect falls and classify movement patterns. The neural network algorithm was created, trained, visualized, and simulated using the Neural Network Toolbox of MATLAB software. Then, the final form of the algorithm was converted into C-code before being uploaded to the microcontroller.

The training input pattern obtained from the sensors is given to the network input unit. The input pattern is propagated forward in the network from layer to layer until the output pattern is generated by the output layer. If the generated output pattern is different from the target output, the error will be determined and propagated backward through the network from the output layer to the input layer. The weights are adjusted as the error is propagated.

The three-layer network shown in Fig. 8 is considered to derive the backpropagation learning law. Let the indices i, j, and k represent the neurons in the input, hidden and output layers, respectively. Generally, i indicates the number of input signals and k denotes the number of recognition patterns. x1, x2, x3 until xn are input signals which propagated forward through the network from the input layer to the output layer, and error signals, e1, e2, e3 until el, propagated backward from the output layer to the input layer. The symbol wij refers to the synaptic weight for the connection between neuron i in the input layer and neuron j in the hidden layer. The symbol wjk represents the synaptic weight between neuron j in the hidden layer and neuron k in the output layer.

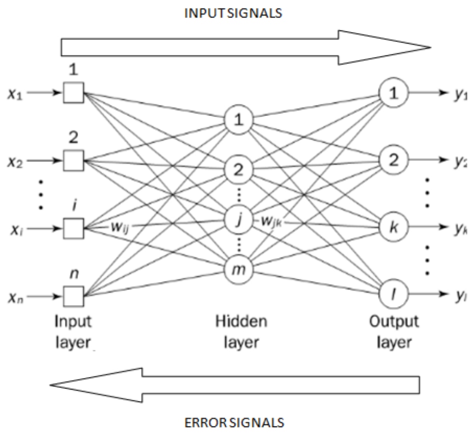


Fig. 8. Multilayer neural network structure

Step 1: Initialization
The synaptic weights and threshold θ of the network are set randomly by constant distribution within a small range:

$$\left(-\frac{2.4}{F_i}, \frac{2.4}{Fi}\right) \quad (3)$$

where F*i* represents the sum of the inputs of neuron i in the network. The weight is initialized based on a neuron-by-neuron basis.

Step 2: Activation
The back-propagation neural network is activated by putting in the inputs $x_1(p)$, $x_2(p)$, $x_3(p)$, until $x_n(p)$ and target outputs $yd_{,1}(p)$, $yd_{,2}(p)$, $yd_{,3}(p)$, until $yd,n(p)$.
(a) The actual outputs of the neurons in the hidden layer, y_j are calculated:

$$y_j(p) = sigmoid[\textstyle\sum_{i=1}^{n} x_i(p) \cdot w_{ij}(p) - \theta_j)] \quad (4)$$

where sigmoid denotes the sigmoid activation function, *n* refers to the number of inputs of neuron *j* in the hidden layer, *p* is the number of iteration and θ is the threshold.
(b)The actual outputs of the neurons, in the output layer, y_k are calculated:

$$y_k(p) = sigmoid[\textstyle\sum_{j=1}^{m} x_{jk}(p) \cdot w_{jk}(p) - \theta_k)] \quad (5)$$

where *m* refers to the number of inputs of neuron k in the output layer and θ is the threshold.

Step 3: Weight training
The weights in the back-propagation network are updated propagating backwards the errors associated with output neurons.

(a) The error gradient, δ_k for the neurons in the output layer is calculated:

$$\delta_k(p) = y_k(p) \cdot \{1 - y_k(p)] \cdot e_k(p) \quad (6)$$

where $e_k(p) = y_{d,k}(p) - y_k(p)$ and $y_{d,k}(p)$ denotes the target output of neuron *k* at iteration *p*.

The weight corrections are calculated:

$$\Delta w_{jk}(p) = \alpha \cdot y_j(p) \cdot \delta_k(p) \quad (7)$$

where *yj* represents the output of neuron *j* in the hidden layer, the symbol $\delta_k(p)$ denotes the error gradient at neuron k in the output layer at iteration p and α is the learning rate.

The weights at the output neurons are updated:

$$\Delta w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \quad (8)$$

(b) The error gradient for the neurons in the hidden layer is calculated:

$$\delta_j(p) = y_j(p) \cdot [1 - y_j(p)] \cdot \textstyle\sum_{k=1}^{l} \delta_k(p) \cdot w_{jk}(p) \quad (9)$$

where *l* refers to the number of neurons in the output layer.
The weight corrections are calculated:

$$\Delta w_{ij}(p) = \alpha \cdot x_i(p) \cdot \delta_j(p) \quad (10)$$

The weights at the hidden neurons are calculated:

$$\Delta w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \quad (11)$$

Step 4: Iteration
The iteration p is increased one by one and then return to Step 2. The process is repeated until the selected error criterion is satisfied.

II. RESULTS AND DISCUSSION

Firstly, the signal profiles from the accelerometers and gyroscopes for both ADL and fall events are presented in Fig. 9. Impulsive motions like jumping and running produce signals almost like a fall motion. This can cause false alarm in the detection system.

Several experiments in a controlled environment were carried out to test and validate the performance and functionality of the system. The experiments aim to debug the program code, fine-tuning and improving the overall performance of the product.

These conducted tests are similar to the previous data collection experiments, where both fall activities and ADL activities simulation are performed by a group of volunteers. Fifteen volunteers are involved in testing this system. Every subject aged ranging from 18 to 40, height ranging from 160cm to 185cm, and weight ranging from 40kg to 90kg. The system is tested against four types of falls and five types of ADL activities. Each activity will be repeated three times by the subjects. Table 3 and Table 4 present the details of the evaluation test 1 and 2.

TABLE 1 Evaluation Test 1 (ADL)

Subjects	15
Ages	18-40
Heights	160-185cm
Activities	Walking, sitting, running, jumping, lying down
No. of times action repeated	3 times
Total ADL performed	(15x 5x3)=225

TABLE 2 Evaluation Test 2 (Fall Activities)

Subjects	15
Ages	18-40
Heights	160-185cm
Activities	Fall forward, fall backwards, fall left, fall right
No. of times action repeated	3 times
Total Fall performed	(15x4x3)=180

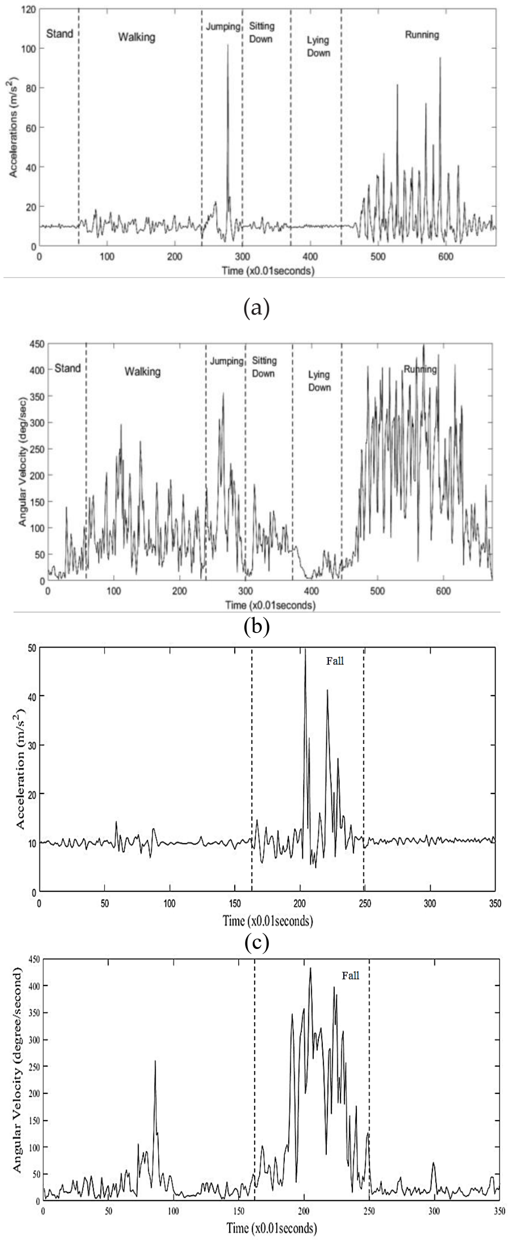


Fig. 9. Sensor signal vs time for (a) ADL accelerometers, (b) ADL gyro (c) a fall event from accelerometers, and (d) a fall event from gyro

The results of the experiments are categorized based on the four possible conditions. The numbers of occurrences for the four conditions are used for the calculations of the accuracy, sensitivity, and specificity of the fall detection algorithm. The four possible conditions for the output of the system are presented below:

- True Positive (TP): the system successfully detects the presence of fall
- False Positive (FP): the system detects a fall, although a fall did not occur;
- True Negative (TN): activity of daily livings (ADL) or non-fall movement is performed and the system does not notify as a fall
- False Negative (FN): the system does not announce a fall although a fall occurs.

The sensitivity, specificity, and accuracy of results are presented to evaluate the performances of the developed system:
Sensitivity represents the ability of the system to detect falls, 100% denoting that all falls are detected.

$$Sensitivity = \frac{TP}{TP+FN} \tag{12}$$

Specificity is the capacity to only detect falls and ignore non-fall events, 100% denoting that no false alarms are announced.

$$Specificity = \frac{TN}{TN+FP} \tag{13}$$

Accuracy is related to the proportion of true results among the population, 100% accuracy denoting 100% sensitivity and specificity.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

Table 5 shows the results of the evaluation test. In general, different motions produce different patterns of signals, which can be used for the classification of fall events. However, the collected signal from the accelerometer and gyroscope are contaminated with noise due to intrinsic and environmental factors. Therefore, a filtering process must be applied to the collected data before applying the fall detection algorithm.

TABLE 3 Validation Results

Total Falls	180	Total ADL	225
True Positive (TP)	173	True Negative (TN)	217
False Negative (FN)	7	False Positive (FP)	8
Sensitivity: 95.5%			
Specificity:96.4%			
Accuracy:96.3%			

By applying the neural network, it is possible to accurately classify fall events and other fall-like activities, which are difficult to distinguish using the threshold method. Due to the similarity of the signal profile, the ADL motion like jumping and running are most likely to produce False Positive results. The accuracy of the neural network system can be improved by including more input patterns in the training phase.

IV. CONCLUSION

This paper proposed a human fall detection system designed using a neural network. As the experimental results indicate, the proposed fall detection method can detect and distinguish between the falling activities and ADL with high sensitivity, high specificity, and high accuracy with values of 95.5%, 96.4%, and 96.3% each. In addition, a vital-sign monitoring device to monitor the body temperature and pulse rate has been successfully developed to improve the emergency alert function of the system. By using the IoT-technology, the system can send alert messages to parents or friends of the users that include important information such as the location and time of the accidents, vital-signs, and nearby hospitals.

In future work, more simulated activities will be added to this system to improve performance. It is the interest of the author to figure out a safe way for the elderly to perform the fall experiments and provide more realistic data. With the technology of fall detection and the Internet of Things, it is hoped that the wearable fall alert system will provide reliable fall detection that can minimize fall injuries for the elderly.

ACKNOWLEDGMENT

The authors would like to thank the *Faculty of Electrical Engineering* of Universiti Teknikal Malaysia Melaka (UTeM) for providing the laboratory facilities and equipment support. The author also would like to thank Intel Malaysia for providing hardware and technical support.

REFERENCES

[1] W. Qu, F. Lin, and W. Xu, "A Real-time Low-complexity Fall Detection System On The Smartphone," in *Connected Health: Applications, Systems and Engineering Technologies (CHASE), 2016 IEEE First International Conference on,* 2016, pp. 354-356.

[2] Mafauzy, Mohamed. "The problems and challenges of the aging population of Malaysia." *The Malaysian journal of medical sciences: MJMS 7.1*, 2000, pp 1-3.

[3] Ramachandran, Anita, and Anupama Karuppiah. "A survey on recent advances in wearable fall detection systems." *BioMed research international 2020*, 2020, pp. 1-17.

[4] Kavya, Thathupara Subramanyan, et al. "Fall Detection System for Elderly People using Vision-Based Analysis." *Science And Technology 23.1, 2020*, pp. 69-83.

[5] Casilari, Eduardo, Raúl Lora-Rivera, and Francisco García-Lagos. "A wearable fall detection system using deep learning." *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems.* Springer, Cham, 2019, pp. 445-456.

[6] N. Anwar and M. Abdullah, "Through-the-wall human sensing based on change detection," in *2015 IEEE International Conference on Imaging Systems and Techniques (IST)*, 2015, pp. 1-5.

[7] Y. Li, K. Ho, and M. Popescu, "Efficient source separation algorithms for acoustic fall detection using a Microsoft Kinect," *IEEE Transactions on Biomedical Engineering,* vol. 61, pp. 745-755, 2014.

[8] A. Kassim, M. S. Jamri, M. S. M. Aras, M. Rashid, and M. Yaacob, "Design and development of obstacle detection and warning device for above abdomen level," in *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*, 2012, pp. 410-413.

[9] B. Aguiar, T. Rocha, J. Silva, and I. Sousa, "Accelerometer-based fall detection for smartphones," in *Medical Measurements and Applications (MeMeA), 2014 IEEE International Symposium on*, 2014, pp. 1-6.

[10] X. Yuan, S. Yu, Q. Dan, G. Wang, and S. Liu, "Fall detection analysis with wearable MEMS-based sensors," in *Electronic Packaging Technology (ICEPT), 2015 16th International Conference on,* 2015, pp. 1184-1187.

[11] G. Shi, J. Zhang, C. Dong, P. Han, Y. Jin, and J. Wang, "Fall detection system based on inertial mems sensors: Analysis design and realization," in *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2015 IEEE International Conference on*, 2015, pp. 1834-1839.

[12] A. T. Özdemir, "An analysis on sensor locations of the human body for wearable fall detection devices: Principles and practice," *Sensors,* vol. 16, p. 1161, 2016.

[13] A. Z. Rakhman and L. E. Nugroho, "Fall detection system using accelerometer and gyroscope based on smartphone," in *Information Technology, Computer and Electrical Engineering (ICITACEE), 2014 1st International Conference on*, 2014, pp. 99-104.

[14] L. N. V. Colon, Y. DeLaHoz, and M. Labrador, "Human fall detection with smartphones," in *Communications (LATINCOM), 2014 IEEE Latin-America Conference on*, 2014, pp. 1-7.

[15] J. Jacob, T. Nguyen, D. Y. Lie, S. Zupancic, J. Bishara, A. Dentino, et al., "A fall detection study on the sensors placement location and a rule-based multi-thresholds algorithm using both accelerometer and gyroscopes," in *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on,* 2011, pp. 666-671.

[16] P. Pierleoni, A. Belli, L. Palma, L. Pernini, and S. Valenti, "A versatile ankle-mounted fall detection device based on attitude heading systems,"

in *Biomedical Circuits and Systems Conference (BioCAS), 2014 IEEE*, 2014, pp. 153-156.

[17] S. Electronics. (2017). *Intel® Edison and Mini Breakout Kit.* Available: https://www.sparkfun.com/products/13025

[18] N. Nuttaitanakul and T. Leauhatong, "A novel algorithm for detection human falling from accelerometer signal using wavelet transform and neural network," in *Information Technology and Electrical Engineering (ICITEE),* 2015 7th International Conference on, 2015, pp. 215-220.

[19] J. L. Chua, Y. C. Chang, and W. K. Lim, "Intelligent visual based fall detection technique for home surveillance," in *Computer, Consumer and Control (IS3C), 2012 International Symposium on,* 2012, pp. 183-187.

[20] M. Vallejo, C. V. Isaza, and J. D. Lopez, "Artificial neural networks as an alternative to traditional fall detection methods," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE,* 2013, pp. 1648-1651.

[21] C. Dinh and M. Struck, "A new real-time fall detection approach using fuzzy logic and a neural network," in *Wearable Micro and Nano Technologies for Personalized Health (pHealth), 2009 6th International Workshop on*, 2009, pp. 57-60.

[22] H. J. Mohd, A. Ismail, T. Ahmad Izzuddin, M. F. Sulaima, and M. S. Mokhtar, "Feasibility study of vehicular heatstroke avoidance system for children," *The International Journal of Engineering And Science,* vol. 4, pp. 14-18, 2015.

[23] T. A. Izzuddin, M. Ariffin, Z. H. Bohari, R. Ghazali, and M. H. Jali, "Movement intention detection using neural network for quadriplegic assistive machine," in *Control System, Computing and Engineering (ICCSCE), 2015 IEEE International Conference on*, 2015, pp. 275-280.

[24] A. Z. Shukor, M. F. Miskon, M. H. Jamaluddin, F. bin Ali, M. F. Asyraf, and M. B. bin Bahar, "A new data glove approach for Malaysian sign language detection," *Procedia Computer Science,* vol. 76, pp. 60-67, 2015.