Applied Computer Science, vol. 16, no. 1, pp. 95–102
doi:10.23743/acs-2020-08

Submitted: 2020-03-14
Revised: 2020-03-20
Accepted: 2020-03-25

remote health monitoring, fall detection, design

Mohanad ABDULHAMID, Deng PETER***

REMOTE HEALTH MONITORING: FALL DETECTION

Abstract

Falling is a serious health issue among the elderly population; it can result in critical injuries like hip fractures. Immobilization caused by injury or unconsciousness means that the victim cannot summon help themselves. With elderly who live alone, not being found for hours after a fall is quite common and drastically increases the significance of fall-induced injuries. With an aging Baby Boomer population, the incidence of falls will only rise in the next few decades. The objective of this paper is to design and create a fall detection system. The system consists of a monitoring device that links wirelessly with a laptop. The device is able to accurately distinguish between fall and non-fall.

1. INTRODUCTION

Healthcare systems in the world have undergone tremendous evolution in the last 50 years. In the early 1960s, we had computers in the form of mainframes being incorporated into healthcare systems. However, there were some problems met from their usage. These mainframes were very few, expensive, large in size and consumed a lot of electrical power and as a result, they had to be shared by several hospitals since independent ownership wasn't feasible an idea to be considered at all (Huang & Newman, 2012).

Come 1970s to early 1990s, there were enormous changes in terms of size and cost for computers & some of the hospital equipment that were invented and innovated at the time. Hospitals were thus able to acquire smaller sized computers and be able to easily operate their independent healthcare systems. Also, in this period, we had the invention of the internet which had a positive impact on health systems. Communication amongst hospitals, their staff and patients greatly improved.

* Al-Hikma University, Karada Kharidge, Baghdad, Iraq, moh1hamid@yahoo.com
** University of Nairobi, P.O. Box 30197-00100, Nairobi, Kenya, researcher12018@yahoo.com

Hospitals were also able to upload and store some of their data (especially patients' data) online so as to make their accessibility easy for authorized personnel (personnel need not be physically present at the premise since all they needed was just a computer, internet connection & and the necessary access password) (Gong, Wang, Zhang & Wang, 2017).

Despite all of the tremendous changes, there was still the issue of affordability of the treatments offered by the hospitals (that were properly equipped with computers & hospital equipment) from the patients' point of view. This had a negative impact on both the hospitals & patients. Hospitals were not getting that large enough a number of patients for treatment while patients were opting for alternatively cheaper treatment plans (which were not that good enough compared to that of properly equipped hospitals) (Saranya, Preethi, Rupasri & Veena, 2018).

Then came the mid-1990s to early-2000 and present where significant technological advancements have taken place. This has seen to great improvements in the healthcare systems with the diversification of remote health monitoring which by definition, is a form of technology which allows a patient to use a mobile medical device to perform tests from outside a clinic and collects the medical and health data to securely transmit to healthcare professionals for remote assessment. As a result, patients, especially the chronically ill, elderly or disabled are able to have increased healthcare access at their homes with decreased healthcare delivery costs (Malasinghe, Ramzan & Dahal, 2019).

The scope of this paper is limited to fall detection and remote viewing of the collected data. Fall detection algorithm is developed and implemented through programming on Arduino Uno board. Remote viewing of the data is done on a laptop after having fall detection data wirelessly transmitted to it.

2. Design and methodology

2.1. The algorithm design

To detect fall along an axis, the acceleration magnitude is considered. This is achieved by a magnitude vector. Consider:

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

where AM is acceleration magnitude.

With the accelerometer output data, the angle change can also be calculated using the dot product. To achieve this, the instantaneous vector and a reference vector are introduced. Instantaneous vector is given by

$$a = (a_x, a_y, a_z) \tag{2}$$

Reference vector is generated when a user stands up. It is given by:

$$b = (b_x, b_y, b_z) \tag{3}$$

Using both the instantaneous vector and reference vector in the following formula:

$$a \cdot b = |a| \cdot |b| \cdot cos\theta \tag{4}$$

Making the angle as subject

$$\theta = cos^{-1}\left(\frac{a \cdot b}{|a| \cdot |b|}\right) \tag{5}$$

In the event of falling, one experiences a momentary free-fall then a large spike in acceleration. In the flow-chart shown in Fig. 1, we have two decision figures: lower threshold AM and upper threshold AM. The algorithm runs in the following manner. First, it checks whether the lower threshold value has been broken by the AM and if so, it then quickly checks whether the upper threshold is broken within a span of 0.5s. If it's not broken, we go back to data collection and if so, the algorithm recognizes this event as a fall. This algorithm's strength is that it requires two AM thresholds to be broken by an activity for a fall to occur.

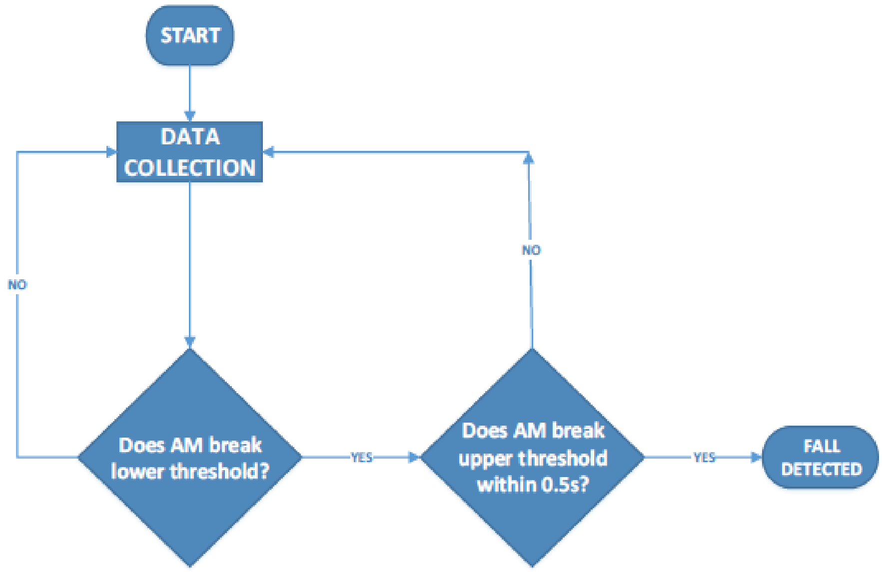

Fig. 1. Fall detection algorithm

2.2. Components

2.2.1. Arduino Uno R3

The Arduino Uno is a very popular board among hobbyists and is the micro-controller board of choice when building small model projects. Because of this, there are extensive tutorials and open source examples available to facilitate learning and familiarizing oneself with the board. In addition to this, we choose this board because of the following characteristics:

1. Operating voltage – The operating voltage of 5V with a 3.3V option is appropriate because both our sensor boards and Bluetooth module operate under 5 or 3.3V power and output readings in the range of 0–5V.
2. Input voltage – The board has a built-in voltage regulator that allows an input voltage range of 7–12V, which is suitable because we plan to power the board with a 9V battery.
3. Memory – The flash memory (32KB) is appropriate because our algorithm programs can be fairly long and require a decent amount of memory on the microcontroller to store them. The Static Random Access Memory(SRAM) (2KB) is a little on the low side, but the algorithms can work around this by not storing too many variables, so as to not exhaust the SRAM capacity.
4. Specialty pins – The Arduino Uno comes with RX/TX pins, which will be used for serial communication with our Bluetooth module. The board also comes with I2C compatible pins, which will be crucial to interface with our digital accelerometer.

2.2.2. Bluetooth module

For wireless data transmission, we choose the HC-05 module which is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. The HC-05 Bluetooth module can be used in a Master or Slave configuration, making it a great solution for wireless communication. We choose this module namely because its pins and power are 5V compatible. It also supports RX/TX serial communication from 9600 to 115200bps (bits per second, baud rate), which makes it fully compatible with our Arduino Uno R3 board.

2.2.3. Sensor

For appropriate fall detection, we choose an accelerometer**.** The model selected is the ADXL345 triple axis digital accelerometer. It has a wide G range (up to ±16g). The range is very wide considering some severe falls are rated at 8 g's. Since it is a digital sensor, the resolution can be adjusted and there is less voltage noise, and less calibration. The ADXL345 gives tri-axial data and requires a minimum of 3.3V power, is I2C compatible and thus our microcontroller board can interface with it correctly.

2. 3. Overall system design

2.3.1. Circuit setup

In order to achieve the assumed goal an appropriate electronic circuit was developed. The overall circuit setup is shown in Fig. 2.

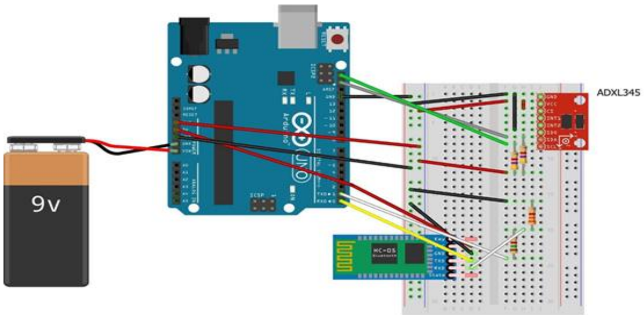

Fig. 2. Overall circuit setup

2.3.2. Block diagram

In the proposed solution, we have the collection of data by the accelerometer, processing of the same by the micro-controller and ascertaining of whether a fall has occurred (Fig. 3 – section A). Our algorithm, is run by the micro-controller. The Bluetooth module receives fall detection data from the micro-controller and transmits the same to a laptop computer where we realize remote viewing of the data. The overall block diagram is shown in Fig.3.

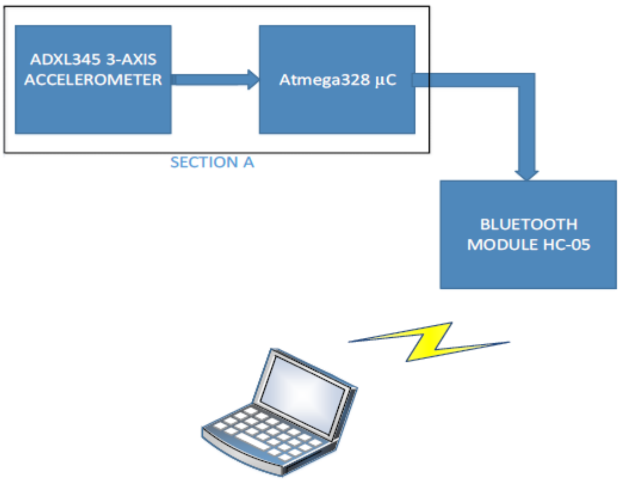

Fig. 3. Overall block diagram

3. RESULTS

The graphs below represent the fall and non-fall data that we collect during our testing phase. They present a comparison between the kind of graphs we expect to see in a fall and non-fall scenarios. From our algorithm, we set the upper threshold value to 2g's so as to ensure that all acceleration values from fall activities break it just as illustrated in Fig. 4. Also, we see that the set lower threshold value of 0.4 g's has been broken. In Fig. 5 and Fig. 6, we have graphs for non-fall activities. In both cases, we see that their acceleration magnitude does not exceed the set upper threshold value of 2g's.

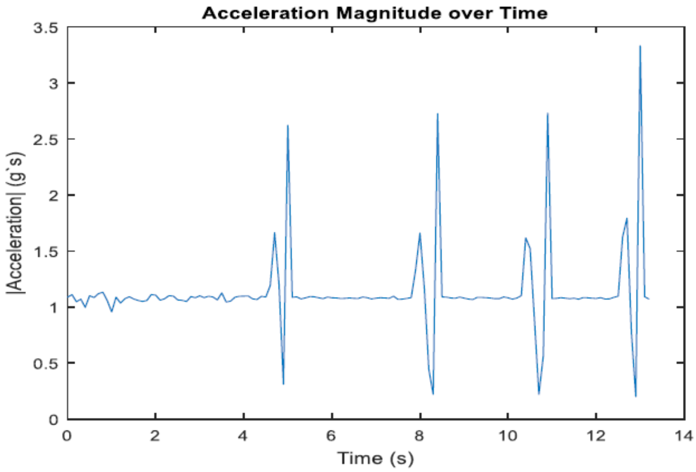

Fig. 4. Graph of falling down

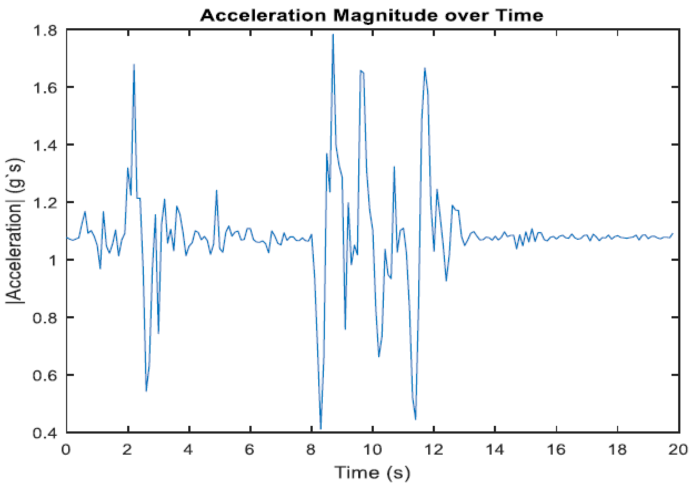

Fig. 5. Standing up and sitting down hard

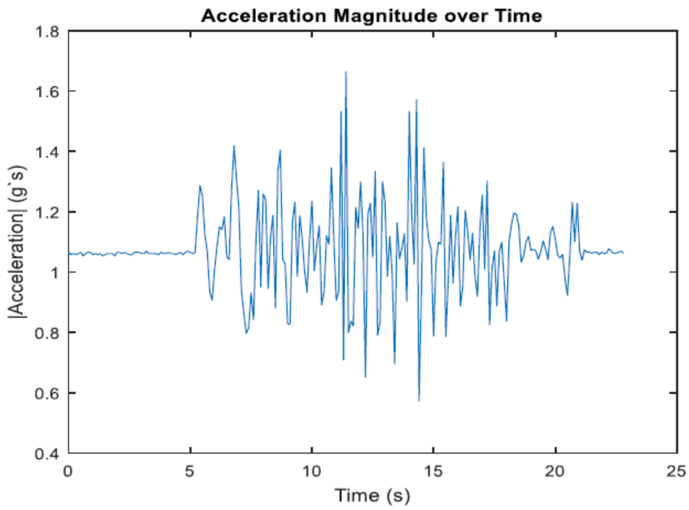

Fig. 6. Walking

4. CONCLUSION

The objective of this paper was to design a fall detection system that links wirelessly with a laptop computer (where we have remote viewing of the collected data). By the conclusion of this paper, we had achieved the primary goal of creating a working system able to recognize falls from non-falls, while wirelessly synched with a laptop.

With this paper, there are some areas for future development. From the commercial point of view, improvements would include: having the system housed in a proper and well-designed casing to prevent its damage in the occurrence of a fall, establishing emergency contacts though the PC-side by sending text messages, reduction of the size of the system by using custom printed circuit boards and Lithium-ion batteries, and porting the PC-side programming onto a mobile phone to realize complete mobile communication. Also, we could add a gyroscope and Global Positioning System(GPS) module to the setup. Both would greatly improve on the overall efficiency of the system

REFERENCES

Gong, S., Wang, Y., Zhang, M., & Wang, C. (2017). Design of remote elderly health monitoring system based on MEMS sensors. In *2017 IEEE International Conference on Information and Automation (ICIA)* (pp. 494–498). Macau: IEEE.

Huang, Y., & Newman, K. (2012). Improve quality of care with remote activity and fall detection using ultrasonic sensors. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 5854–5857). San Diego, CA: IEEE.

Malasinghe, L., Ramzan, N., & Dahal, K. (2019). Remote patient monitoring: a comprehensive study. *Journal of Ambient Intelligence and Humanized Computing*, 10(1), 57–76.

Saranya, M., Preethi, R., Rupasri, M., & Veena, S. (2018). A survey on health monitoring system by using IOT. *International Journal for Research in Applied Science & Engineering Technology*, 6(III), 778–782.