# Independent Living of Elderly Senior using Power Efficient and Interrupt- Driven Algorithm

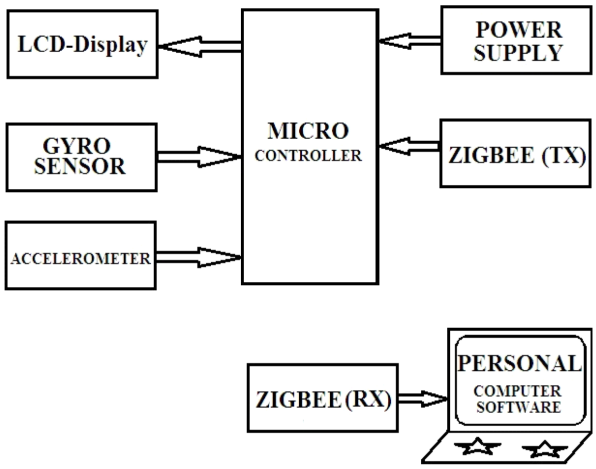
[1.] S. Bharathi, [2.] B. Vinothini, [3.] V. Lavanya
St.Anne`S College Of Engineering And Technology
Department Of Electronics And Communication Engineering

***Abstract:*** **This project presents the real-time activity recognition and fall detection system. It is tuned for robustness and real-time performance by combining human-understandable rules and classifiers trained with machine learning algorithms. The system consists of two wearable sensors. an accelerometer and a gyroscope, placed on the abdomen and the right thigh. The recognition of the user's activities and detection of falls is performed on a laptop using the raw sensors' data acquired through Bluetooth. The offline evaluation of the system's performance was conducted**

## I. INTRODUCTION

FALLS pose major health problems for people aged 65 years and over. Falls occur in 30–60% of older adults each year, and 10–20% of these result in injury, hospitalization and/or death [1]. Without immediate help, seniors may suffer pain, emotional distress or even develop other complications including pneumonia, dehydration, hypothermia [2]. There-fore, immediate help after fall is of critical importance as it could lower the risk of complications and death, and greatly increase the likelihood of returning to independent living. In recent years, MEMS (microelectromechani cal systems) inertial sensors have sparked an intense interest in studying falls.The world's population is aging rapidly, threatening to overwhelm the society's capacity to take care of its elderly members.The percentage of persons aged 65 or over in developed countries is projected to rise from 7.5% in 2009 to 16% in 2050Fall detection is an important component of many ambient assisted living systems because approximately half of the hospitalizations of the elderly are caused by falls. The architecture of the system combines rules to recognize postures (static activities), which ensure the behavior of the system is predictable and robust, and classifiers trained with machine learning (ML) algorithms, to recognize dynamic activities, for which the rules are not sufficiently accurate.For the Fall detection, rules are used that take into account high accelerations associated with falls and the recognized horizontal orientation (e.g., falling is often followed by lying).

## II.BLOCK DIAGRAM




A single integrated circuit containing a processor core, memory, and microcontroller is a small computer on a programmable input/output peripherals. It act as brain of the system.

- A gyroscope is a device for measuring or maintaining orientation, based on the principles of angular momentum.
- An accelerometer is a device that measures acceleration relative to a free-fall
- A liquid-crystal display (LCD) is electronic visual display that uses the light modulating properties of liquid crystals. Liquid crystals do not emit light directly.
- A regulated power supply is an circuit that converts unregulated AC into a constant DC. This DC is used to power the circuits.
- Zigbee is a wireless device that is used for communication between UART supported devices

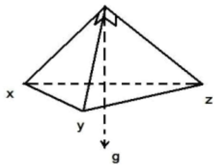
## III.ACCELEROMETERS


### .A. Hardware and Software Platform

The hardware platform for the proposed algorithms is shown in Fig. 1. It is designed to be a wrist-worn wearable device (WD) and consists of a CC2430 chip, a LED, a buzzer and an ADXL345 accelerometer. CC2430 is an enhanced 8051 MCU with a built-in ZigBee Transceiver from Texas Instruments.

## IV. FALL DETECTION ALGORITHM

The proposed fall detection algorithm is derived from an algorithm [26] proposed in an application note by Ana-log Devices. The original algorithm characterizes a fall




accelerometer reading by all axes converging towards zero. This pattern can be exactly captured by using FREE_FALL interrupt.

proposed 0.75 g and 30 ms for THRESH_FF and TIME_FF which are threshold and time windows of FREE_FALL interrupt respectively. There is a serious flaw with this threshold. ADXL345 asserts FREE_FALL when all axes are smaller than THRESH_FF for a time longer than TIME_FF. This assertion is not based on the vector sum of all

axes which is $sv = x^2 + y^2 + z^2$. There is a possibility that the accelerometer is positioned such that all axes have equal angles with the gravity vector.

### B. Algorithm Description

The proposed algorithm is modeled as a finite-state machine. A flow chart shown in Fig. 3 is used to illustrate the algorithm. It essentially consists of six states, from F0 to F5. The details of this algorithm are described as below.

(F0) F0 is the initial state as well as reset state. ADXL345 is initialized as follows:

Data rate: 25 Hz. ACTIVITY, INACTIVITY are enabled and mapped to INT1. ACTIVITY threshold is 2.25 g. INACTIVITY threshold is 0.5 g and its detection time window is 1 second.

Link mode is enabled. By enabling link mode in the state F0, INACTIVITY will not be asserted repeatedly if seniors are not moving (e.g. sleeping).

FREE_FALL is enabled and mapped to INT2. FREE_FALL threshold is 0.5625 g and its detec-tion window is 20 ms. FIFO is initialized to TRIGGER mode and is triggered by FREE_FALL. When triggered, FIFO will hold the latest 16 samples before the trigger, discard earlier ones, continue to collect until full. When returning from other states, CC2430 timer ticks (started in F1) will be stopped and the system

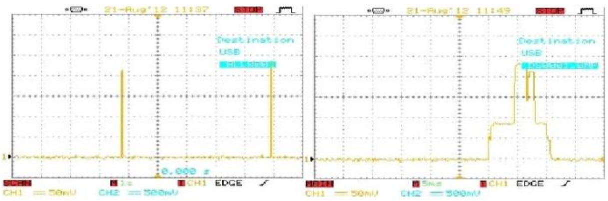
*Experimental Studies*

*Trial Study:* A trial study has been carried out to evaluate fall detection accuracies in real world scenarios at Jurong Central Daycare Centre, which is a typical eldercare It is observed that during continuous walking, falls were never be falsely detected as INACTIVITY interrupt did not have a chance to assert. While sitting down, wrists occa-sionally hit the desk with high impact, a false positive fall will be accidentally triggered. It can also be observed that young subjects easily created false positives (3/20). In contrast, seniors in their real daily lives only creates 1.153 times per day as evaluated in Section IV-C.1. A very probable reason is that young stronger subjects create large impacts in their ADLs which were easily confused with a fall in the fall detection algorithm, The total current consumption of a WD is measured by connecting a 7.5_ resistor in series with it. The WD polls message from its parent range extender every 5 seconds. Fig. 5a shows a measurement of power consumption of the WD for a duration of 10 seconds when statically placed. Two current peaks spaced by 5 seconds can be easily identified. Each peak corresponds to CC2430 waking up from sleep mode for polling. The magnitude of the current peak is aapproximately 4.4 × 50 *mV* /7.5_ = 29.33 *m A*. The sleep current is too small to be readable. Instead, it was measured by a multimeter to be 69 $\mu$ *A*. Within the 69 $\mu$ *A* sleeping current, 40 $\mu$ *A* is attributed to the current consumption of ADXL345 at an ODR (output data rate) of 25 Hz. The remaining 29 $\mu$ *A* is consumed by CC2430 in sleep mode and its peripherals.

TABLE I

SIMULATED FALL RESULTS

| Activity type | With a fall | Falls detected |
|---|---|---|
| (1) | yes | 19/20 |
| (1) | no | 0/20 |
| (2) | yes | 19/20 |
| (2) | no | 0/20 |
| (3) | yes | 17/20 |
| (3) | no | 0/20 |
| (4) | no | 3/20 |



(a) (b)

activity five times for two minutes each time. Then activities (1), (2), (3) were performed again for five times, but each with a fall in the middle. Test results are shown in Table It is observed that during continuous walking, falls were never be falsely detected as INACTIVITY interrupt did not have a chance to assert. While sitting down, wrists occa-sionally hit the desk with high impact, a false positive fall will be accidentally triggered. It can also be observed that young subjects easily created false positives (3/20). In contrast, seniors in their real daily lives only creates 1.153 times per day as evaluated in Section IV-C.1. A very probable reason is that young stronger subjects create large impacts in their ADLs which were easily confused with a fall in the fall detection algorithm, as discussed in Section .

*D. Battery Life Analysis*

The total current consumption of a WD is measured by connecting a 7.5_ resistor in series with it. The WD polls message from its parent range extender every 5 seconds. Fig. 5a shows a measurement of power consumption of the WD for a duration of 10 seconds when statically placed. Two current peaks spaced by 5 seconds can be easily identified. Each peak corresponds to CC2430 waking up from sleep mode for polling. The magnitude of the current peak is aapproximately $4.4 \times 50\ mV/7.5\_ = 29.33\ mA$. The sleep current is too small to be readable. Instead, it was measured by a multimeter to be 69 $\mu A$. Within the 69 $\mu A$ sleeping current, 40 $\mu A$ is attributed to the current consumption of ADXL345 at an ODR (output data rate) of 25 Hz. The remaining 29 $\mu A$ is consumed by CC2430 in sleep mode and its peripherals.

## V. CLASSIFICATION OF ACTIVITIES OF DAILY LIVING

In the proposed fall detection algorithm, depending on the wrist's movement, the algorithm flows dynamically among the six states. The time it spends in each state gives a lot of information about activities of a wearer.

Traditionally, studies of ADLs such as Wockets aim to identify a number of activities including walking, climbing stairs, sitting, sleeping, bathing, cooking, using multiple sen-sors. In this paper, since only one accelerometer is used, only a few simple activities can be inferred from accelerometer data. The identifiable activities are "Walk", "Random" (random wrist movements), and "Quiet" (no movement at all). These activities are purely inferred from accelerometer information. However, it is possible to categorize these activities into more specific activities or even abnormalities by using accelerometer data along with additional contextual information such as time and location. For instance, "Quiet" at mid night most likely means sleeping. Too many "Random" incidences during night might indicate poor sleeping quality. "Quiet" in bath rooms longer than certain time (e.g., 30 minutes) is abnormal and an alert can be raised.

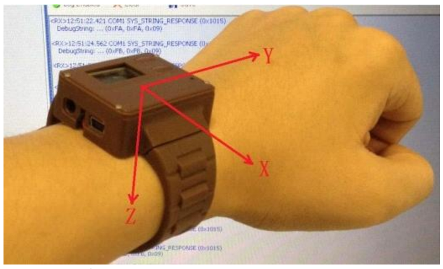



Fig. 6. Alignment of accelerometer axes when worn by a human subject.

obtained by experiments. Both interrupts need to operate in AC mode and link mode is still enabled. By setting both thresholds to the same small value, when a wrist moves a small amount, ACTIVITY will be asserted, and when the wrist is relatively static, INACTIVITY will be asserted in 1 second. This setting only lives within state F0. Whenever the fall detection algorithm transits to state F1, both thresholds will be immediately configured to 2.25 g and 0.5 g for ACTIVITY and INACTIVITY respectively.

With above hardware settings, four types of events can be identified from the fall detection algorithm and they are named with the following conventions:

$E0$ : generated whenever an INACTIVITY interrupt asserted in state F0
$E1$ : generated whenever an ACTIVITY interrupt asserted in state F1
$E2$ : generated whenever the fall detection algorithm transits from state F0 to state F1
$E3$ : generated whenever the fall detection algorithm transits from state F1 to state F2

There are underlining physical meanings associated with these four events. $E0$ is generated whenever a wrist changes from active movements to motionless. $E1$ is generated when the wrist slightly moves or changes orientation. This cor-responds to casual movements human makes during ADLs. $E2$ corresponds to weightlessness at the wrist, which is generated when the wrist is lowered from a higher level. This happens when the subject falls, walks (swings arms) and sits down. During ADLs, accelerometer values will typically fluctuate around the gravity value. Capturing only the weight-less part is good enough to estimate the activity level. $E3$ is typically generated during drastic movements such as a fall, suddenly sitting down and putting down arms onto desks.

Along with above events, directions of Y-axis during the occurrence of these events are also taken as a feature. Fig. 6 shows the alignment of axes of the accelerometer when worn by a human subject on his left wrist. Y-axis is always aligned with the subject's lower arm. Y-axis information is critically important for estimating ADL as Y-axis is typically pointing approximately downwards when the human subject is standing or walking. During sleeping, Y-axis is usually approximately horizontal.

*B. Preprocessing*

Fig. 7 shows raw data collected from a WD worn by a human subject who performed two actions including "Walk" and "Random" ("Quiet" is not shown as it basically contain
no events). Each action lasted exactly a minute. Below is a list of intuitions based on the raw data.
During "Random", only $E0$ and $E1$ are generated in alternating manners (as the accelerometer's link mode is enabled).During "Walk", as the arm swings, FREE_FALL interrupt will be generated and the fall detection algorithm moves from state F0 to state F1. As there is no impact following FREE_FALL during walking, the fall detection algorithm returns to state F0 shortly. This pattern repeats and generates a lot of $E2$ as well as $E0$ and $E1$ along the way. Also observe that Y-axis values are always positive as Y-axis typically points downwards during walking. During "Quiet", no events will be generated. Leaves more energy at the present time than an event which occurred a long time ago. The host MCU could tell if an event has just occurred recently by evaluating current energy level of this event.

The proposed ADL classifier takes the second approach. The first approach requires uncertain amount of memory space to keep track of all events in the past 10 seconds, which should be avoided in an 8-bit MCU. In addition, it will compute the same event multiple times as the 10-second time-window shifts by one each time. The second approach can be easily implemented and requires less computational resources. For each of the four events ($E0$, $E1$, $E2$ and $E3$), an energy function is proposed to keep track of the activeness of this event in the recent past.

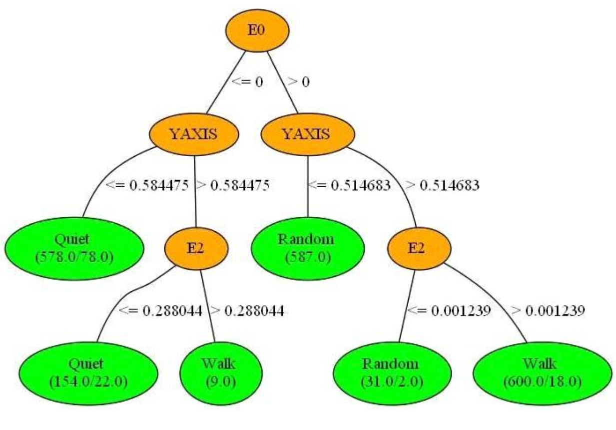
The contribution of an event to its energy function decays with time. An exponential decay function is proposed in Equation 2.

$$\eta(t) = e^{-t/\tau} . \tag{2}$$

where $t$ is the elapsed time and $\tau$ = 10 seconds which means that an event's energy decays to $e^{-1} \approx 0.37$ of its original value after 10 seconds.

Upon the occurrence of an event, energy of all events in the past are reduced by a factor $\eta$ ($0 < \eta \leq$ which is related to the elapsed time of the previous event. The energy of the occurred event will be incremented correspondingly. The exponential function for computing the decay factor can be efficiently approximated by using a look-up table and interpolation in host MCUs.

The proposed ADL classifier is completely event-driven as it only computes upon occurrence of an event. This leads to a problem when a WD is "Quiet". Consider a human subject is walking initially and suddenly keeps still. An event $E0$ will be generated one second after he keeps still. At point, the classification of his action over the past 10 second could be "Walk". As no event will be generated after this $E0$, the algorithm will not compute at all thereafter and the classification result remains as "Walk" which does not reflect the fact. To resolve this issue caused due to no events ever being generated during "Quiet", an artificial event is created and named as $E_{-1}$ to signify its special purpose. Upon occurrence of an $E_0$ event, after which might follow a long "Quiet" period, a dummy check event is scheduled to be set in 5 seconds using a timer. Any other event which occurs before this timer expires will cancel this timer. If no other event cancels, the algorithm, upon seeing this dummy event, will reset energy functions of all events to zero.




Classification Outcome

Decision tree learning has been extensively used in statis-tics, data mining and machine learning. C4.5, developed by

Ross Quinlan, is a well-known learning algorithm for gener-ating decision trees. As it accepts numerical attributes, data obtained from the Algorithm 1 can be conveniently fed into a C4.5 training algorithm. This paper uses a C4.5 implementa-tion in Weka (Waikato Environment for Knowledge Analysis), a data mining suite developed at the University of Waikato.

### *D. Classification Algorithm*

Six groups of ADL data have been collected from six young human subjects. Each subject performed 4 minutes of “Walk”, 4 minutes of “Quiet” and 5 minutes of “Random”. Four out of the six groups of ADL data are used training data and the other two are used as testing data.

Details of the classification results for one of the testing subjects along the time line. Activity of the first 240 seconds is “Walk”, followed by another 240 seconds of “Quiet” and ended by 300 seconds of “Random”. It can be seen that the classifier performs fairly well. In fact, it is robust enough that in the “Random” period, the subject pauses a few seconds in between. Thus, the labels for that interval is actually incorrect. Nevertheless, the classifierrobust

## VI. CONCLUSION

In this paper, a fall detection algorithm and an ADL classifi-cation algorithm for a wrist-worn device have been proposed.They are interrupt-driven and can be efficiently implemented in battery-powered MCUs with limited clock speed and RAM. Both algorithms are hardware-dependent as they are based on a modern digital MEMS accelerometer which supports various interrupts and FIFO. The advantage is that they are more power-efficient than conventional algorithms which must examine and process each sample of accelerometer data. By processing accelerometer data completely locally, a WD does not have to stream massive sensor data out wirelessly thus saving both power and bandwidth. In the context of *e-Guardian*, less bandwidth consumption results in better scalability thus allowing a single system to accommodate more WDs.

## REFERENCES

- L. Z. Rubenstein, “Falls in older people: Epidemiology, risk factors and strategies for prevention,” Age Ageing, vol. 35, no. 2, pp. 37–41, Sep. 2006.
- Lifeline With Autoalert Option. [Online]. Available: http:// www.lifelinesys.com/content/lifeline-products/auto-alert,accessed Jul. 14, 2013.
- F. Sposaro and G. Tyson, “iFall: An Android application for fall monitoring and response,” in Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC), Sep. 2009, pp. 6119–6122.