A Patient-Specific Single Sensor IoT-Based Wearable Fall Prediction and Detection System

Wala Saadeh, *Member, IEEE,* Saad Adnan Butt, *Student Member,* and Muhammad Awais Bin Altaf, *Member, IEEE*

***Abstract*— Falls in older adults are a major cause of morbidity and mortality and are a key class of preventable injuries. This paper presents a patient-specific (PS) fall prediction and detection prototype system that utilizes a single tri-axial accelerometer attached to the patient's thigh to distinguish between activities of daily living (ADL) and fall events. The proposed system consists of two modes of operation: 1) fast mode for fall predication (FMFP) predicting a fall event (300msec-700msec) before occurring, 2) slow mode for fall detection (SMFD) with a 1-sec latency for detecting a fall event. The nonlinear Support Vector Machine Classifier (NLSVM)-based FMFP algorithm extracts 7 discriminating features for the pre-fall case to identify a fall risk event and alarm the patient. The proposed SMFD algorithm utilizes a Three-cascaded 1-sec sliding frames classification architecture with a linear regression-based offline training to identify a single and optimal threshold for each patient. Fall incidence will trigger an alarming notice to the concern healthcare providers via the internet. Experiments are performed with 20 different subjects (age above 65 years) and a total number of 100 associated falls and ADL recordings indoors and outdoors. The accuracy of the proposed algorithms is furthermore validated via MobiFall Dataset. FMFP achieves sensitivity and specificity of 97.8% and 99.1%, respectively, while SMFD achieves sensitivity and specificity of 98.6% and 99.3%, respectively, for a total number of 600 measured falls and ADL cases from 77 subjects.**

***Index Terms*— Fall detection, fall prediction, feature extraction, gait monitoring, patient-specific, support-vector machine, threshold detection, wearable sensor.**

I. Introduction

The rising populace of elderly people has increased the risk of accidental and unassisted fall events [1]. Accidental falls are a major concern for the elder people; being the main cause for hospitalization and the second leading cause of unintended injury-related demises among the elder people in the world [1]-[3]. According to the U.S. Centers for Disease Control and Prevention, 25% of the Americans aged 65 or above experience a fall once each year [4]. Every 11 second an elderly person is treated in the hospital for a fall and, every 19 seconds an elderly person dies from a fall. These falls cause more than 27,000 deaths and more than 800,000 injuries annually and cost tens of thousands million dollars every year [1], [2]. Fig. 1 shows the number of reported fatal fall events and their expenses per age group for elder people (> 65 years) in the USA only in 2010 [3].

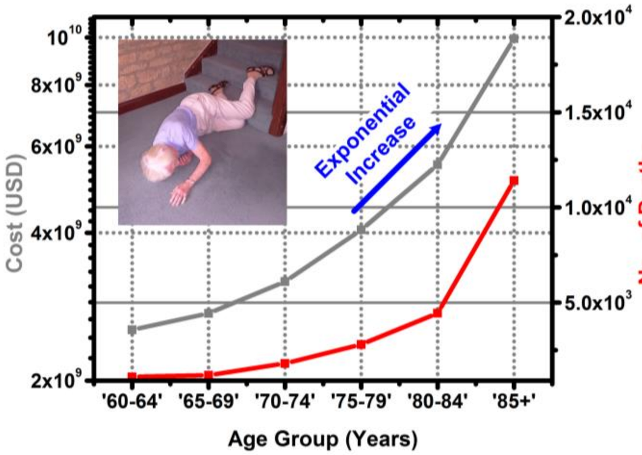


Fig. 1: Distribution of fatal fall events and cost per age group in the USA in 2010 [3].

The undesirable effects of fall events have led to wide attention in the fall risk assessments, detection, and prediction systems by health-care professionals. It is crucial for the health-care providers to determine circumstances and scenarios that led to a fall event and advise a mechanism to mitigate such falls [5]. In practice, most of the fall risk assessment are usually collected through patient's interviews and questionnaires, fall diaries, phone calls, and simple physical performance tests [6]. Data collection through the formerly mentioned methods provides pertinent information, but these statistics cannot be treated always as the reliable medical record since elderly people often forget or fail to recall the precise circumstances of their fall event [5], [6]. Identifying the person as a high fall-risk patient is not sufficient to protect the patient. Therefore, accurate continuous monitoring with a decentralized on-spot decision [7] - [9], and recording mechanism is critical for the fall-prone patients and elderly people [10]-[13].

Several research studies have proposed different methods to detect a fall event, however, very few ones predicted the fall event before it occurs [10]-[15]. The study in [6] presents an algorithm to classify elder women subjects as high or low fall risk patients. Ref. [15] proposes a fall prediction and detection algorithm based on a tri-axial accelerometer, predicting the fall event 200~400msec before the collision with a limited number of test cases. Falls can be detected by monitoring a person's surrounding using a fixed video camera with computer vision [14]-[17], wearable cameras [18], pressure sensors [19], smart tiles [20], and acoustic sensors (Microphone Array System) [21],

Manuscript submitted for review on October 04, 2018. This work was funded by the Lahore University of Management Sciences (LUMS), Lahore, Pakistan under startup grant number STG-EED-1216.

W. Saadeh, S. Butt, and M. A. B. Altaf are with the Electrical Engineering Department, Lahore University of Management Sciences, Lahore 54792, Pakistan (e-mail: wala.saadeh@lums.edu.pk).

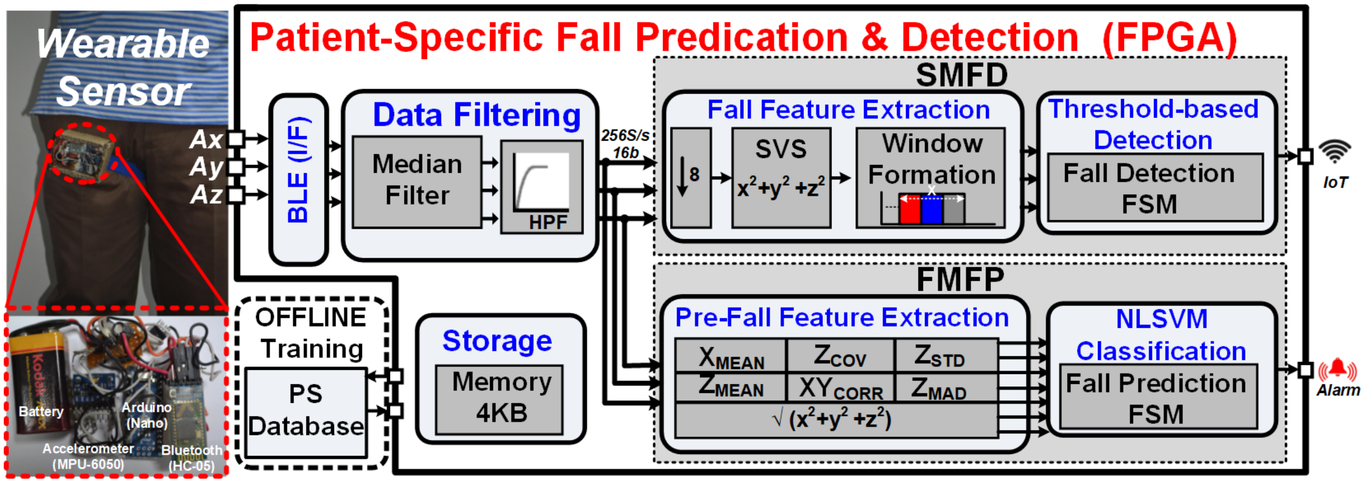


Fig. 2: Block diagram of the proposed Fall Prediction and Detection Prototype System.

Radar-based systems [22], and Microsoft Kinetics [23]. The alternative approach is to utilize inertial wearable sensors such as a gyroscope and an accelerometer [24], [25] or the inertial sensors of the mobile-phones [10], [26]-[27]. However, the gyroscope is typically not preferable due to higher power consumption than the accelerometer [11]. To continuously monitor fall events, it is more suitable to use wearable devices with detection and prediction capabilities as they can be utilized anywhere and not limited to certain locations [24].

Several fall detection algorithms recognize multiple features to differentiate between fall cases and regular activities of daily living (ADL). These features include falling speed, stride time, acceleration coordinates, posture information, inactivity periods, and angular velocity [2], [25], [28]. The extracted features are then classified as a fall event or an ADL, using a threshold value or by combining posture information and kinematic thresholds, to generate a decision [11]. Nevertheless, the falling speed and gait features exhibit deviation from person-to-person and varies with aging too [29],[30]. Therefore, it is essential for a fall detection system to tackle these differences for accurate fall detection [31]. Currently, the fall detection systems attempt to minimize the false positives; where some ADLs are incorrectly identified as fall events, at the cost of long latency and enhanced computational cost [11], [32]. Moreover, current fall-detection systems do not have a low power storage to record the fall's acceleration data [1].

This paper presents a patient-specific (PS) fall prediction and detection prototype system. In this work, a single tri-axial accelerometer sensor attached to a person's thigh is utilized to predict and detect the fall events and save their data for detailed follow-up by the medical practitioner. The patient will be notified if a fall event is predicated while the occurrence of a fall event will initiate an alarming notice to the concern healthcare providers by connecting it through the internet. The performance of the proposed system is verified on 77 subjects with 600 recordings including 100 recordings of Fall/ADLs from 20 elder subjects (aged above 65 years).

The structure of the paper is as follow. Section II describes the proposed PS fall prediction and detection system. Section III details the feature extraction. Section IV elaborates the proposed fall prediction algorithm, and Section V explains the fall detection algorithm. Section VI demonstrates the measurement results and discussion. Finally, Section VII concludes the paper.

II. Patient-Specific Fall Prediction and Detection System

The proposed fall prediction and detection prototype system comprised of two parts; the data acquisition part followed by the PS classification part. The block diagram of the proposed system is presented in Fig. 2. In the sensing part, a tri-axial accelerometer is used to extract the acceleration of the elderly person in three orthogonal directions, the X, Y, and Z-axis, at a sampling rate of 256 S/sec. The acquired accelerometer data (Ax, Ay, and Az) along X-, Y-, and the Z-axis, respectively, are then transferred via low energy Bluetooth interface (BLE (I/F)) to the classification part that is implemented on a Field-Programmable Gate Array (FPGA).

A. *Wearable Sensing Part*

The system board is designed with an accelerometer sensor, an Arduino (Nano) microcontroller, a 16-bit ADC, and Bluetooth (HC-05), powered by a 9V battery with an overall size of 6 cm × 3.5 cm × 2 cm. The sensor chosen is the MPU-6050 Tri-Axial accelerometer with an adjustable full-scale range of ±2g, ±4g, ±8g, and ±16g, which can be attached comfortably to the patient's thigh without distressing the person's routine life activities. The upper thigh location is the preferred location for placing the sensor since it connects the bottom kinetic chain (legs and feet) to the parts responsible for keeping steadiness (core and head) [6]. In addition, elder people with chances of high fall risk show less harmonic acceleration ratio sequence in the pelvis and upper thigh [6], [33]-[34].

The ideal wearing position of the wearable sensor is shown in Fig. 3 (a) on the subject's thigh. The tri-axial accelerometer sensor is mounted inside the wearable part where the Y-axis of the sensor is always in parallel with human body Y-axis. However, both X-axis and Z-axis of the sensor can be misaligned from the human body X-axis and Z-axis, respectively, due to wearing the device in a different angle $(\theta \neq 0)$ as depicted in Fig. 3(b). Both X-axis and Z-axis will be rotated by the same angle θ forming two new axes X' and Z'. The rotation will lead the sensor to read acceleration in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSRE.2019.2911602, IEEE Transactions on Neural Systems and Rehabilitation Engineering

these different axes X' and Z'. These new axes are related to ideal axes as given in both (1) and (2):

$$X' = X\,cos(\theta) \qquad (1)$$

$$Z' = Z\,cos(\theta) \qquad (2)$$

Ideally, we want $\theta = 0$ where $(cos\ (0) = 1)$. For optimal functionality of the proposed system, the rotation angle θ should be maintained less than 20° $(cos\ (20) = 0.939)$.

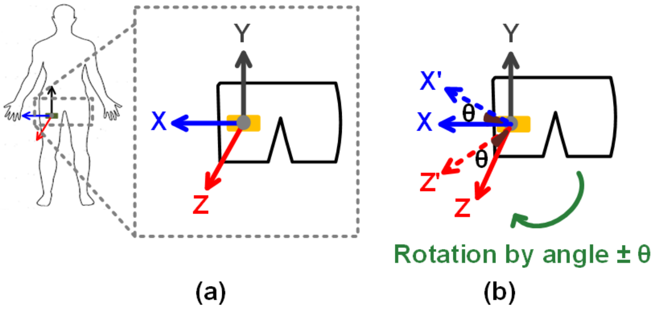


Fig. 3: (a) Ideal wearing position for the wearable sensor where the sensor axes are aligned with patient body axes, (b) Rotation of the wearable sensor forming new axes in X' and Z' directions.

B. *Fall Prediction and Detection System*

The acquired acceleration data along each axis is processed through a median filter followed by a high pass filter (HPF) to mitigate the in-band noise. The HPF eliminate the gravity acceleration since the tri-axial accelerometer generates an output that is a combination of the desired dynamic acceleration and gravity acceleration [25]. The system works in two parallel modes: a fast mode for fall prediction (FMFP) and a slow mode for fall detection (SMFD). The FMFP operates on the incoming acceleration data at 256 S/sec while for SMFD the data are sub-sampled to 8 S/sec. This is because, for the fall prediction, it is required to regularly check (every 100 msec) for potential fall event to take timely actions while for the fall detection; it is required to look at longer period to identify a fall event (3 sec in this system). If a large sampling rate to be used for fall detection too, extra overhead of unnecessary computation will be performed which will consume more power. In each operating mode, the feature extraction block extracts the selective features for the fall and forms a feature vector (FV). A 4KB storage records the accelerometer data of a fall event for further examination by physicians. If any fall event is predicted, it will alarm the person to take precautionary actions while if a fall event is detected, it will be transferred via the internet to the health care providers to initiate an immediate help for the fallen elderly person.

III. Feature Extraction

A. *Fall Prediction Features*

In the FMFP, the feature extraction block extracts the discriminating features from the tri-axial accelerometer data for fall prediction. In order to identify the most appropriate minimum set of features for fall prediction, several features are examined for highest sensitivity and specificity on 600 patients' recordings from our experiments and MobiFall dataset as shown in Table I [6], [27]. The discriminating features for the FMFP are carefully selected to achieve the highest sensitivity and specificity based on features utilized in a recent study conducted in [6]. The selected seven features provide the highest sensitivity and specificity of 97.8% and 99.1%, respectively. Most of these features can be also utilized for assessing patients as high or low fall risk patients [6]. The adopted set of features for fall prediction includes 1) mean acceleration of X-axis(μ_x), 2) mean acceleration of Z-axis (μ_z), 3) standard deviation of acceleration of Z-axis (σ_z), 4) coefficient of variance of Z-axis (COV_z), 5) correlation coefficient between X-axis and Z-axis $(CORR_{xz})$, 6) mean amplitude deviation of X-axis (MAD_x), and 7) the total sum vector (SV). COV_z is defined as the ratio between the standard deviation σ_z and the mean μ_z:

$$COV_z = \frac{\sigma_z}{\mu_z} \qquad (3)$$

COV_z is utilized to detect the variability in relation to the mean of acceleration events. $CORR_{xz}$ is used to assess the strength and direction of the relationships between X-axis and Z-axis accelerations and is defined as follows [35]:

$$CORR_{xz} = \frac{n(\sum x.z) - (\sum x).(\sum z)}{\sqrt{(\ \ n\sum x^2 - (\sum x)^2) - (n\sum z^2 - (\sum z)^2)}} \qquad (4)$$

Where n is the number of samples in the window, x and z are the filtered acceleration data in X and Z directions, respectively.

MAD_x describes the mean distance of acceleration data points about the mean [35]:

$$MAD_x = \frac{1}{n}\sum |x_i - \mu_x| \qquad (5)$$

Where n is the number of samples in the window, x_i the i^{th} filtered acceleration sample in the X direction within the window and μ_x is the mean resultant acceleration value of the window.

The general definition of a total sum vector (SV) magnitude for the 3-D acceleration can be found by [25]:

$$SV = \sqrt{x^2 + \ y^2 + z^2} \qquad (6)$$

Where x, y, and z (g) denote the filtered accelerations along X-, Y-, and Z-axis, respectively.

B. *Fall Detection Features*

The fall event of a person can be identified as an unintentional sudden change from the upright/straight position to the resting or lying position. Therefore, when a person falls, the body's acceleration suddenly increases. In the proposed SMFD algorithm, only 1 feature is utilized to detect the fall, the total sum vector square (SVS) is computed by finding the square of the SV as given by [10]:

$$SVS = x^2 + y^2 + z^2 \qquad (7)$$

The proposed definition of SVS compared to SV will expand the acceleration data range and simplifies defining the threshold value for each subject as computing square root reduces the range of the output data [10].

Table I: List of tested features for fall prediction with corresponding sensitivity and specificity.

	Features Set	Sensitivity	Specificity
1	X_MEAN, Z_MEAN, X_COV, X_SMA, X_PFREQ, X_ENERGY, SV	89.3%	87.2%
2	X_MEAN, Z_COV, X_SMA, X_RMS, X_ENERGY, SV	85.5%	82.6%
3	Y_PFREQ, Z_COV,Y_MCR, Y_RMS, Y_STD, Y_COV, SV	87.0%	88.9%
4	Z_COV, Z_MEAN, Z_MAD, Z_STD, Z_SMA, SV	92.6%	95.4%
5	Z_COV, XY_CORR, X_MEAN, X_MEAN, Z_STD,Z_MAD, SV	91.3%	92.6%
6	X_MEAN,Z_MEAN, Z_STD, Z_COV, XZ_CORR, Z_MAD, SV	97.8%	99.1%
7	Z_COV, Z_MEAN, Z_MAD, Z_STD, Z_SMA, SV	96.8%	97.5%
8	Z_COV, Z_MEAN, Z_MAD, Z_STD, Z_SMA, SV	93.1%	96.8%

PFREQ: peak frequency SMA: signal magnitude area
STD: standard deviation MCR: mean crossing rate
CORR: correlation coefficient between two axes

IV. Proposed Fall Predication Algorithm

Nonlinear Support Vector Machine Classifier (NLSVM) is utilized in FMFP to classify the incoming features as pre-fall or normal ADL events. Fig. 4 shows the flowchart of the NLSVM-based fall prediction algorithm. Initially, the digitized acceleration data will be passed to the feature extraction engine at every 100msec to extract the discerning features. In the case of a new patient, the NLSVM PS parameters will be weighed by utilizing an off-line NLSVM learner [36]. Off-line learning module based on MATLAB Statistics and Machine Learning Toolbox [37], [38] needs a FV and former fall information for each particular patient. The trained PS parameters are recorded in the off-line PS repository to be utilized by on-system NLSVM classifier. The trained parameters are then uploaded on-processor using serial peripheral interface (SPI) protocol. The NLSVM classifier will be ready to process the runtime FVs after the PS parameters are uploaded in the system repository. If the patient's parameters are previously placed in the database, the learning phase of the classifier will be by-passed. The predicted fall event by NLSVM classifier will initiate an alarm to the patient. To minimize the false alarms, three consecutive pre-fall decisions from NLSVM classifier are required to identify the case as a real pre-fall case and assign Fall Risk=1. The number of pre-fall decisions for the NLSVM is optimized to obtain maximum prediction accuracy and maximum pre-fall time. The Fall Risk alarm will be kept high for 3 sec (minimum) such that it can be seen by the SMFD block. In the proposed algorithm the fall event will be predicated before (300msec-700msec) of occurring as compared to 200msec-400msec in [15].

The proposed NLSVM classification is selected as an optimal choice among different machine learning algorithms, i.e., Linear Support Vector Machine (LSVM), Decision Tree (DT), and K Nearest Neighbor (KNN)). Fig. 5 shows the Receiver Operating Characteristics (ROC) of different machine-learning classifiers along with their Area under Curve (AUC). ROC curves were obtained using MATLAB Statistics and Machine Learning Toolbox [38] by feeding 7 features for pre-fall predication and looking for 3 consecutive decisions of the classifier. If all of the 3 decisions are showing pre-fall case, then the final output of the classifier will be also decided as a pre-fall case. Since the validation was based on the MobiFall dataset [27], the classifier output and known outcome available from the database are compared to compute the accuracy of the system. NLSVM achieves the highest classification accuracy compared to DT, LSVM, and KNN.

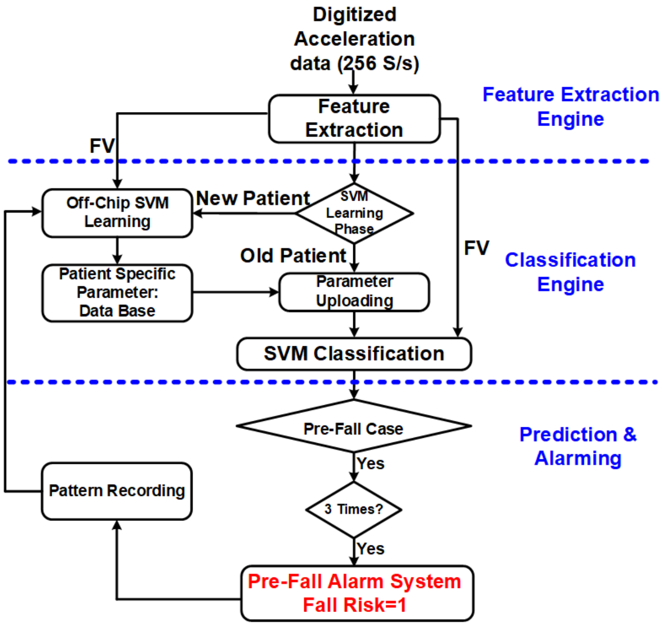


Fig. 4: Flowchart of the fall prediction algorithm.

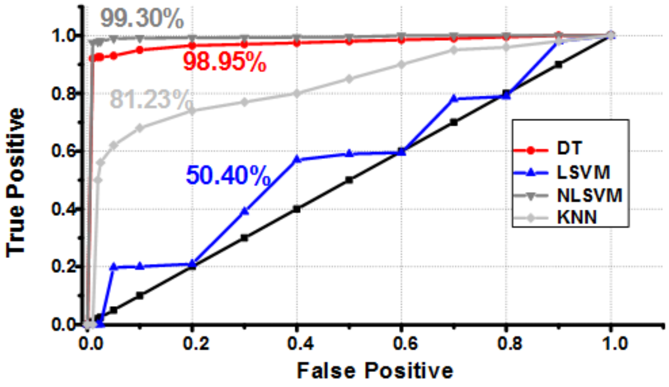


Fig. 5: ROC of different machine-learning classifiers along with their achieved classification accuracy.

The NLSVM classification operates in two stages, learning and decision phases. During the learning phase, the classifier is trained for a specific subject with the corresponding PS parameters. The NLSVM classification is realized using a Radial basis function (RBF). The RBF-NLSVM boundary equation is defined as in (8) [36].

$$\text{RBF} = \sum_{i=1}^{N} (\alpha_i * \exp(-0.5 * (\|X_i - X_C\|^2)/\arg^2)) \quad (8)$$

Where N is the number of support vectors (SVs), arg defines the alteration of RBF, α is PS vector, and Xi and Xc are the FV and SV, respectively, during the learning process. Xi and Xc are both 7-dimension vectors. N, arg, α, and Xc are PS parameters identified during the learning phase. During the classifying phase, the real-time inward FVs (Xi) is delivered to the NLSVM predictor with the formerly calculated PS parameters and then the fall event is predicted based on (9) [36].

$$\sum_{i=1}^{N}(\alpha_i * \exp(-0.5 * (\|X_i - X_C\|^2)/arg^2)) = \begin{pmatrix} > 0 & \text{Pre} - \text{Fall} \\ < 0 & \text{No Fall} \end{pmatrix} \qquad (9)$$

In order to compare the system performance and hardware resources single SVS feature FMFP versus the proposed 7 features FMFP, Table II is constructed. It shows the comparison of synthesized results based on CMOS 0.18um process of the proposed seven-feature FV and single-feature FV i.e. SVS, with NLSVM as prediction classifier. The single-feature FV implementation will reduce the area and energy utilization by 38.3% and 28.1%, respectively, but at the cost of deteriorated prediction sensitivity and specificity of 73% and 81%, respectively, which is relatively below acceptable range for a fall prediction system. The prediction time (latency) will remain the same as 300-700msec due to the utilization of parallel computation of all features in the proposed seven features implementation.

Table II: Resources and accuracy comparison between single feature FMFP and the proposed 7 features FMFP.

	SVS + Feature Prediction	Proposed 7 Features + Prediction	
Area* (gates)	29.8K	49.3K	↑ 38.3% ☹
Energy*	97 nJ	135 nJ	↑ 28.1% ☹
Sensitivity	73%	97.8%	↑ 33.9% ☺
Specificity	81%	99.1%	↑ 22.2% ☺

* Synthesized using CMOS 0.18um process

V. Proposed PS-Threshold Based Fall Detection Algorithm

For distinguishing a fall event from an ADL, a fall event is divided into three stages, i.e., the pre-fall phase, the fall impact, and the post-fall phase. The FV formation is formed based on tracking the SVS values that occur in 3 consecutive frames, where Ei_j is an indication of the SVS value in each frame, 'i' is the frame number 1, 2 or 3 and 'j' is the iteration number, shown in (5).

$$Xj = [\, E1_j \;\; E2_j \;\; E3_j \,] \qquad (10)$$

Where X_j is the FV. These FVs will be used to identify the fall events. They are explained as follows:

A. Pre-Fall Phase: A fall normally occurs when a person is carrying out a regular movement such as walking or sitting which can be labeled by low to medium acceleration values that will mark the beginning of a fall case. During this phase, the SVS typically have low values before the fall. Therefore, the first base for distinguishing a fall is to have all the instances of SVS values to be less than a predefined PS threshold (PS_TH). If this condition is achieved, a value of '0' will be stored in E1 and '1' otherwise. The PS_TH needs to be updated for each patient.

B. Fall Impact: During this phase, a person will experience a rapid falling of the body toward the ground, ending with a vertical shock on the ground. The duration of this high acceleration spike is (300–500msec) [32]. This is shown as a high amplitude in the acceleration curve with $SVS > PS\_TH$. Hence, the second base for distinguishing a fall case is a large spike exceeding the PS_TH in the second time frame with E2 of '1'.

C. Post-Fall: In general, a person, after falling and hitting the ground or any solid surface, remains in an immobile situation for a small duration (minimum of "1-sec" time frame) before moving again [32]. This is identified by an intermission of the horizontal line where SVS < PS_TH which is the third base for identifying a fall event with E3 of '0'.

The three above conditions represent the entire FV, which is exploited in the proposed PS fall detection algorithm. The FV is a 3-bit code based on the result of each frame $[E_1\ E_2\ E_3]$. Four different configurations of frame designs are investigated to find the ideal frame length that achieves the optimum trade-off between detection accuracy, complexity and system latency. Fig. 6 summarizes the comparison for different windowing configurations. These frame designs are described as 1) a 2 sec non-overlapped frame; 2) a 1 sec non-overlapped sliding frame; 3) a 2 sec overlapped frame with 0.5 sec augmentation, and 4) a 1 sec overlapped frame with 0.5 sec augmentation. Analytical results demonstrate that case 1) has the lowest hardware complexity but with low performance (detection accuracy of 75% and the highest system latency (2 sec)). Both cases of 3) and 4) have small latency (0.5 sec) and medium performance (detection accuracy of 85% and 87%, respectively) but with a huge cost for hardware implementation. Whereas, case 2) achieves medium latency (1 sec), and high performance (detection accuracy of >95%) with relaxed hardware implementation cost. The FV evaluation through the non-overlapping frame will result in reduced hardware resources compared to the overlapping one, for example, 2) will consume 27% fewer resources compared to 4) due to a) increase in the storage requirement and b) enhanced clock frequency for the overlapping data. Therefore, a frame size of 1 sec is selected in our proposed design and the FV is updated after each 1 sec with the new incoming acceleration data. The fall event is also detected after 1 sec from its occurrence.

Most of the conventional systems identify a fall as having higher acceleration than ADLs that exceeds a threshold value [25]. However, it is unreasonable to consider a constant acceleration value as a threshold above which all motions are considered as a fall and vice versa. Because this value may differ for people in different situations, health conditions and ages [32]. Selecting a low threshold may increase false positives during ADL, resulting in lower specificity. On the contrary, too stringent threshold selection may lead to failure of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSRE.2019.2911602, IEEE Transactions on Neural Systems and Rehabilitation Engineering

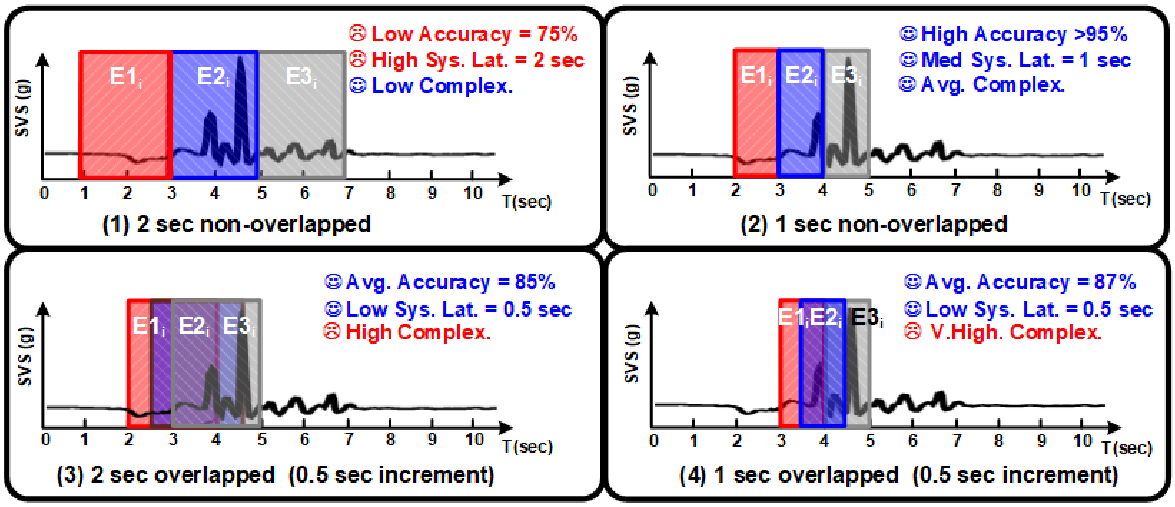


Fig. 6: Different confiurations for sliding frame FV formation for fall detection (a) 2 sec non-overlapped frame, (b) 1 sec non-overlapped frame, (c) 2 sec frame with 0.5 sec increment and, (d) 1 sec frame with 0.5 sec increment.

the fall detection algorithm. Fig. 7 shows the SVS for two patients for a fall event (front fall on knees) and an ADL case (ascending stairs). Subject A is 30 years old male with a height of 177 cm and weight of 102 Kg, while Subject B is a 26 years old female with a height of 170 cm and weight of 90 Kg. From Fig. 7 (a) and (c), Subject A needs to have a high PS_TH to detect fall case correctly (~1.0), while Subject B in Fig. 7 (b) and (d) needs a smaller value of PS_TH (~0.75). Selecting the same threshold value for both subjects will erroneously detect an ADL as a fall or vice versa. The PS_TH value is determined once for each patient off-chip. The PS threshold (PS_TH) is computed off-line with the help of MATLAB and uploaded through SPI protocol to the FPGA. In this work, linear regression is utilized to determine the optimal PS_TH to perform on-sensor processing and reduce the Bluetooth data overhead. Each subject's PS_TH need to be determined, therefore, the training set is required.

The flow chart of the proposed fall detection algorithm is presented in Fig.8. Fall detection decision formation is based on the 3-bits FV formed by the 3 consecutive "1-sec" frames. A fall event is identified by a '010' code as it resembles a high amplitude in the central frame only. For the cases where the last two bits are '01', they can be fall events. Hence, the verdict is delayed until the next cycle after receiving the adjacent 1-sec time frame of the acceleration data. In rare cases where a fall event produces two spikes larger than the PS_TH value in two consecutive 1-sec frames, the SMFD system will check if there was a fall event predicated by FMFP to decide a fall or no fall case. The coordination between the two modes of the systems will make the overall system more reliable and reduce false negatives. The remaining cases are considered ADLs since the conditions for fall event are not met. If a fall event is identified, the tri-axial acceleration information is kept in the local repository and an alarming signal is transmitted to the health care provider through the internet. The fall will be spotted after 1 sec of the occurrence which is >50% less latency than [1],[2],[23].

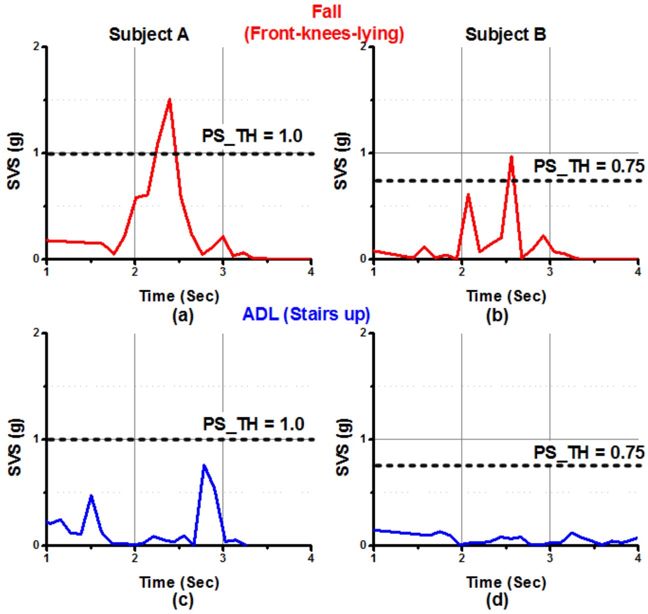


Fig. 7: SVS for two subjects for a fall and no fall cases (a) and (b) are the fall cases for patients A and B, respectively, (c) and (d) are the ADL case for patients A and B, respectively.

VI. Measurement Results And Discussion

To verify the working of the proposed system in the real environment, it was tested on elder subjects. Despite the non-invasiveness nature of the proposed system, the Institutional Review Board (IRB) approval was processed along with the participant consent form under the IRB Organization number LUMS-IRB20170114 for the device testing in human subjects. All the falls and ADLs are performed in a realistic way similar to the method utilized in MobiFall dataset [27]. However, we used a thicker mattress (10 cm) to dampen the fall and protect the elder subjects who participated in the experiments. The Fall/ADL validation experiments are performed by 20 different subjects (12 males and 8 females) (age: 65-70) (Body Mass

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSRE.2019.2911602, IEEE Transactions on Neural Systems and Rehabilitation Engineering

Index (BMI): 28.3-35.1). The validation experiments combined with the MobiFall dataset comprise of 6 different fall cases and 11 multiple ADLs from a total of 77 subjects with greater than 600 recorded trials. The subjects in the MobiFall dataset are healthy 42 males (age: 20-47 years, BMI: 19.78-34.68) and 15 females (age: 20- 36 years, BMI: 18.36-31.14) [27]. The ADLs are selected based on their consistency and resemblance to realistic falls that may increase the false positives. Explicitly, they are chosen established on the subsequent norms: a) fall-like events where the subject typically stays immobile at the end, for example, stepping in a car or sitting on a seat; b) ADL like standing, walking, and descending and ascending the stairs; c) abrupt and rapid events that are alike to fall events, such as jogging and jumping. The four falls comprise a) forward lying fall, b) back chair fall, c) front knees fall, and d) side fall.

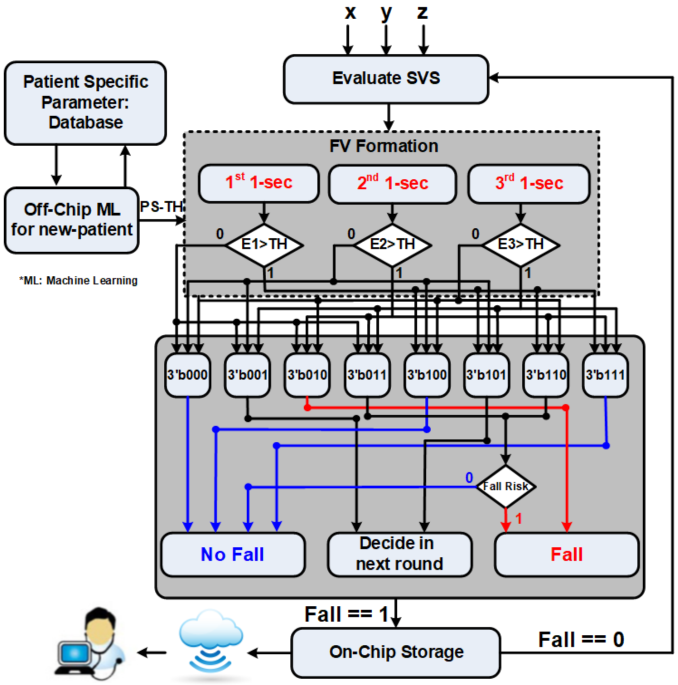


Fig.8: Flowchart of the proposed fall detection algorithm.

Fig.9 presents the measurement results of one healthy subject (Male, 65 years, BMI of 28.3) performing four distinct tasks (jumping, sitting on the seat, falling on the right side, and falling on the front side). The fall prediction features are evaluated every 100msec (not shown in Fig. 9) and fed into NLSVM classifier for pre-fall decision as described in Eq. (9). If a pre-fall event is detected, a Fall Risk alarm will be set to 1 for 3 sec. The SVS feature for fall detection is shown in Fig. 9. For both cases of ADL in Fig.9 (a) and (b), the fall risk remains as 0 while performing these normal daily activities which means that no fall is predicted. However, in the cases of fall events as shown in Fig.9 (c) and (d), the fall event is predicted as fall risk alarm raised to 1 with pre-fall impact time of 400msec and 600msec, respectively. From the offline training of the proposed SMFD system, PS_TH is computed to be 1.5 for this subject. Each case in Fig.9 resulted in a unique 3-bit output FV that is passed to the detection algorithm for fall detection. The proposed SMFD algorithm correctly identifies jumping and sitting on a seat as No Fall and front knee- fall and falling on right sideways as fall cases.

Fig. 10 (a) shows the logic resources utilized by the proposed SMFD/FMFP system (feature extraction and classification) on FPGA Virtex5 (XC5VLX110T), which result into equivalent logic cells of 62.3K [39]. Power budget distribution of the sensor, microcontroller, and FPGA for the proposed system is shown in Fig. 10 (b). The power distribution chart focuses mainly on processing power with SMFD/FMFP consumes 0.34mW at an operating clock of 1 KHz, and does not include wireless transmission power, which will dominate the power budget.

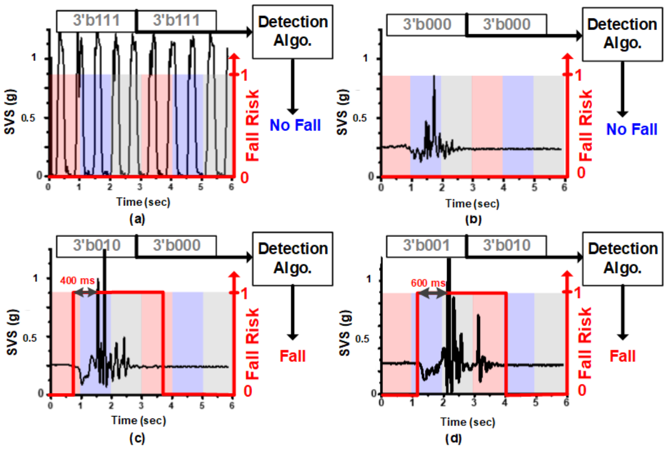
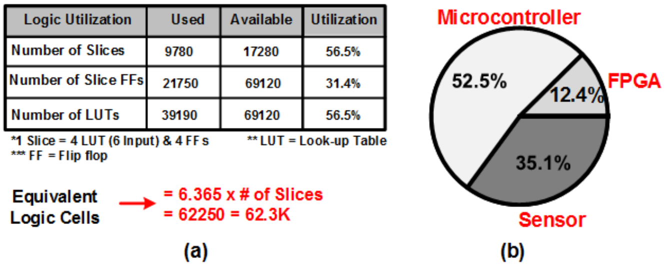


Fig.9: Application of different scenarios for fall detection algorithm on one subject (male, 65 years, and BMI of 28.3) (a) Jumping, (b) Sitting on a chair, (c) Fall on the right side, and (d) Forward from standing, use of hands to dampen fall.



Logic Utilization	Used	Available	Utilization
Number of Slices	9780	17280	56.5%
Number of Slice FFs	21750	69120	31.4%
Number of LUTs	39190	69120	56.5%

*1 Slice = 4 LUT (6 Input) & 4 FFs ** LUT = Look-up Table
*** FF = Flip flop

Equivalent Logic Cells → = 6.365 x # of Slices
= 62250 = 62.3K

Fig.10: a) FPGA resource utilization for the proposed SMFD/FMFP system, b) Power distribution of the overall system.

To compare the performance of the proposed FMFP and SMFD algorithms confusions matrix are formed as shown in Fig. 11. The evaluation shows that, for 600 tests including measured experiments and MobiFall dataset results, the proposed FMFP algorithm accomplishes sensitivity and specificity of 97.8% and 99.1%, respectively, with maximum prediction time of 700 msec before the fall impact.

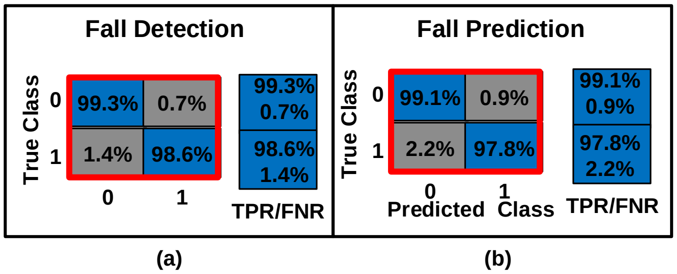
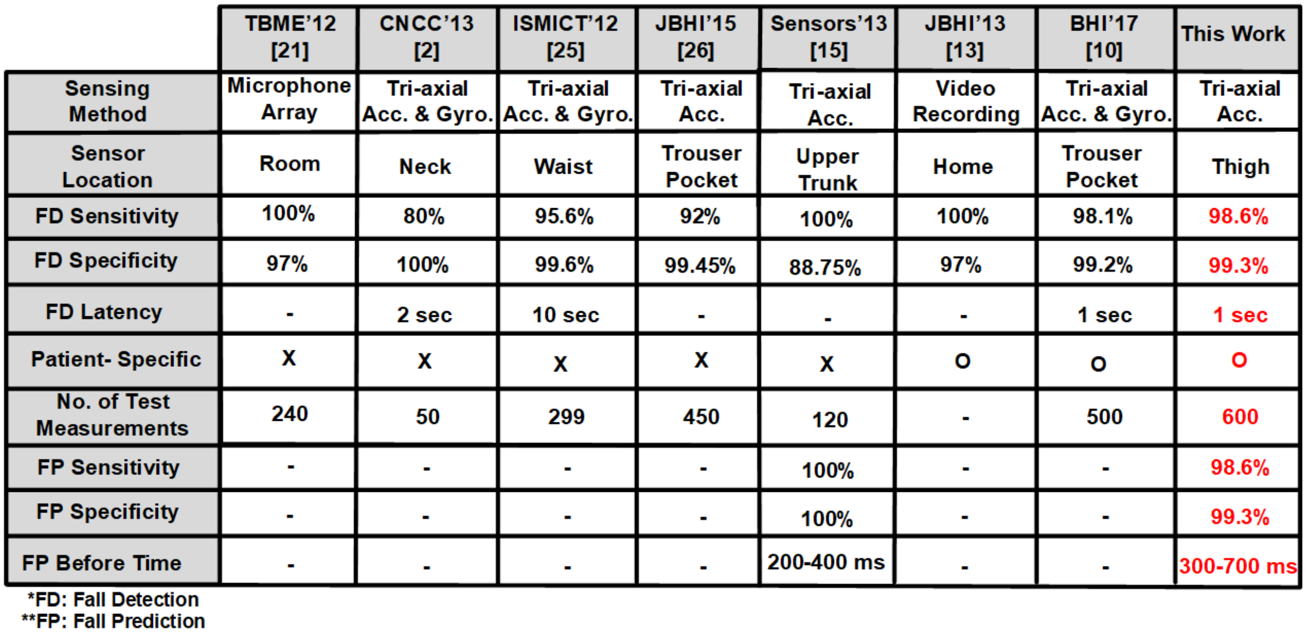


Fig. 11: Confusion Matrix of the proposed fall a) detection and b) prediction.

Table III. Comparison between proposed algorithm and state-of-art work.



	TBME'12 [21]	CNCC'13 [2]	ISMICT'12 [25]	JBHI'15 [26]	Sensors'13 [15]	JBHI'13 [13]	BHI'17 [10]	This Work
Sensing Method	Microphone Array	Tri-axial Acc. & Gyro.	Tri-axial Acc. & Gyro.	Tri-axial Acc.	Tri-axial Acc.	Video Recording	Tri-axial Acc. & Gyro.	Tri-axial Acc.
Sensor Location	Room	Neck	Waist	Trouser Pocket	Upper Trunk	Home	Trouser Pocket	Thigh
FD Sensitivity	100%	80%	95.6%	92%	100%	100%	98.1%	98.6%
FD Specificity	97%	100%	99.6%	99.45%	88.75%	97%	99.2%	99.3%
FD Latency	-	2 sec	10 sec	-	-	-	1 sec	1 sec
Patient- Specific	X	X	X	X	X	O	O	O
No. of Test Measurements	240	50	299	450	120	-	500	600
FP Sensitivity	-	-	-	-	100%	-	-	98.6%
FP Specificity	-	-	-	-	100%	-	-	99.3%
FP Before Time	-	-	-	-	200-400 ms	-	-	300-700 ms

*FD: Fall Detection
**FP: Fall Prediction

The proposed SMFD algorithm achieves sensitivity and specificity of 98.6% and 99.3%, respectively. It achieves a small latency of 1 sec for fall detection, which is >50% less time than [1], [2], [23]. It achieves superior performance than the multi-sensor [2] and multi-threshold [25] acquisition and algorithms, respectively. Table III compares the proposed algorithms with the state-of-the-art works on fall-prediction and detection. Ref [15] achieves slight better sensitivity and specificity compared to the proposed FMFP but does not report latency of the system and utilizes the Hidden Markov Model which will be hardware costly compared to the proposed implementation. Moreover, the reported prediction accuracy is based on 8 young people's simulated activities and the accuracy will be severely affected if tested in a real-time environment due to the utilization of non-PS approach. Offline training for NLSVM parameters and PS-threshold findings reduces the hardware complexity and high power utilization burden on the wearable device system.

VII. Conclusion

In this work, we propose a PS single sensor fall prediction and detection prototype system. It will alarm the patient to take action if a fall case is predicted while if a fall event is detected, it will be communicated to health care providers. The detected fall events will be alarmed to the health care providers through Clouds to provide immediate help to the fallen elder person. The accuracy of the proposed algorithms is validated via MobiFall Dataset and our performed experiments. FMFP achieves sensitivity and specificity of 97.8% and 99.1%, respectively, while SMFD achieves sensitivity and specificity of 98.6% and 99.3%, respectively, for a total 600 measured ADL and falls from 77 subjects. The performance can be improved by studying further falls and fall-like events.

References

[1] D. Chen, *et al.*, "A wearable wireless fall detection system with accelerators," in Proc. *IEEE Int. Conf. Robot. Biomimet. (ROBIO)*, Apr. 2011, pp. 2259-2263.

[2] W. Baek, *et al.*, "Real life applicable fall detection system based on wireless body area network" in *Proc. IEEE Consum. Commun. Network. Conf. (CCNC)*, Jan. 2013, pp. 62–67.

[3] Injury Prevention and Control: Data and Statistics (WISQARSTM), [Online]. Available: http://www.cdc.gov/injury/wisqars/

[4] "Falls Prevention Facts", [Online] Available, https://www.ncoa.org/news/resources-for-reporters/get-the-facts/falls-prevention-facts/

[5] Y.S. Delahoz, *et al.*, "Survey on Fall Detection and Fall Prevention Using Wearable and External Sensors," *Sensors*, vol. 14, no. 10, pp. 19806–19842, Oct. 2014.

[6] A. Hua, *et al.*, "Accelerometer-based predictive models of fall risk in older women: a pilot study," *Nature Digital Medicine*, 2018.

[7] S. Abubakar, *et al.*, "A Wearable Auto-Patient Adaptive ECG Processor for Shockable Cardiac Arrhythmia," *IEEE Asian Solid-State Circuits Conf. (ASSCC)*, Nov. 2018, pp. 267-268.

[8] W. Saadeh, *et al.*, "A 1.1mW Ground Effect-Resilient Body-Coupled Communication Transceiver with Pseudo OFDM for Head and Body Area Network," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 52, no. 10, pp. 2690- 2702, Oct. 2017.

[9] S. Abubakar *et al.* "A Wearable Long-Term Single-Lead ECG Processor for Early Detection of Cardiac Arrhythmia," Proc. *IEEE Design Automation and Test in Europe (DATE)*, pp. 961-966 Mar. 2018.

[10] W. Saadeh, *et al.*, "A High Accuracy and Low Latency Patient-Specific Wearable Fall Detection System," *IEEE Biomedical and Health Informa tics (BHI)*, 2017, pp. 441-444.

[11] S. Abbate, *et al.*, "Recognition of false alarms in fall detection systems," in Proc. *IEEE Int. Work. Consum. eHealth Platforms, Services, and Applications, CCNC* Workshop, Jan. 2011, pp. 538–543.

[12] M. Altaf, *et al.*, "A 0.21µJ Patient-Specific REM/Non-REM Sleep Classifier for Alzheimer Patients," in *Proc. IEEE Biomedical Circuits and Systems (BioCAS)*, Oct. 2017, pp. 652-655.

[13] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE J. Biomed. Health Informat*., vol. 17, no. 3, pp. 579–590, May 2013.

[14] B. Mirmahboub, *et al.*, "Automatic monocular system for human fall detection based on variations in silhouette area," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 2, pp. 427–436, Feb. 2013.

[15] L.Tong, *et al.*, “HMM-Based Human Fall Detection and Prediction Method Using Tri-Axial Accelerometer,” *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849-1856, May 2013.

[16] E. Auvinet, *et al.*, “Fall detection with multiple cameras: An occlusion-resistant method based on 3-D silhouette vertical distribution,” *IEEE Trans. Inf. Technol. Biomed*., vol. 15, no. 2, pp. 290–300, Mar. 2011.

[17] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, “Robust video surveillance for fall detection based on human shape deformation,” *IEEE Trans. Circuits Syst. Video Technol*., vol. 21, no. 5, pp. 611–622, May 2011.

[18] K. Ozcan, *et al.*, “Autonomous Fall Detection With Wearable Cameras by Using Relative Entropy Distance Measure,” *IEEE Trans. Human-Machine Interface sys.* vol. 47, NO. 1, pp. 31-39, Feb. 2017

[19] H. Rimminen, *et al.*, “Detection of falls among the elderly by a floor sensor using the electric near field,” *IEEE Trans. Inf. Technol. Biomed*., vol. 14, no. 6, pp. 1475–1476, Nov. 2010.

[20] M. Daher, *et al.*, “Elder Tracking and Fall Detection System Using Smart Tiles,” *IEEE Sensors Journal*, vol. 17, no. 2, pp. 469-479, Jan. 2017

[21] Y. Li, *et al.*, “A microphone array system for automatic fall detection,” *IEEE Trans. Biomed. Eng.*, vol. 59, no. 5, pp. 1291–1301, May 2012.

[22] C. Garripoli, *et al.*, “Embedded DSP-Based Telehealth Radar System for Remote In-Door Fall Detection,” *IEEE J. Bio. Health Inform. (JBHI)*, vol. 19, no. 1, pp. 92-101, Jan. 2015.

[23] E. E. Stone, *et al*., “Fall Detection in Homes of Older Adults Using the Microsoft Kinect,” *IEEE J. Bio. Health Inform. (JBHI)*, vol. 19, no. 1, pp. 290-301, Jan. 2015.

[24] Q. Li, *et al.*, “Accurate, Fast Fall Detection Using Gyroscopes and Accelerometer-Derived Posture Information,” *IEEE Int. Work. Wear. Implant. Body Sensor Networks*, Jun. 2009, pp. 138-143.

[25] A. Sorvala, *et al.*, “A two-threshold fall detection algorithm for reducing false alarms,” *IEEE Int. Symp. Med. Info. Commun. Techn. (ISMICT)*, May 2012, pp. 1-4

[26] L.-J. Kau, and C. Chen, “A Smart Phone-Based Pocket Fall Accident Detection, Positioning, and Rescue System,” *IEEE J. Bio. Health Inform. (JBHI)*, vol. 19, no. 1, pp. 44-56, Jan. 2015,

[27] G. Vavoulas, *et al*., “The MobiFall Dataset: Fall Detection and Classification with a Smartphone”, *Int. J. Monit. Surveill. Techn. Research*, pp 44-56, 2014.

[28] W. Saadeh, et al., “A Wearable Neuro-Degenerative Diseases Classifier System based on Gait Dynamics,” *IEEE VLSI-SoC*, Oct. 2017, pp. 1-6.

[29] S. Ko, *et al*., “Age-Associated Differences in the Gait Pattern Changes of Older Adults during Fast-Speed and Fatigue Conditions: Results from the Baltimore Longitudinal Study of Ageing,” *Age and Ageing*, vol. 39, no.6, pp. 688–694, 2010.

[30] R. Bartlett, “Introduction to Sports Biomechanics: Analyzing Human Movement Patterns”, 2nd Edition, Routledge, 2007

[31] B. J. Fregly, *et al*., “Design of patient-specific gait modifications for knee osteoarthritis rehabilitation,” *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 9, pp. 1687-1695, Sept. 2007.

[32] M. Yu, *et al.*, “An online one class support vector machine-based person-specific fall detection system for monitoring an elderly individual in a room environment,” *IEEE J. Biomed. Health Informatics*, vol. 17, no. 6, pp. 1002–1014, Nov. 2013.

[33] H. B Menz, *et al.*, “Acceleration patterns of the head and pelvis when walking are associated with risk of falling in community-dwelling older people,” *J. Gerontol. A. Biol. Sci. Med. Sci*., vol. 58, no. 5, pp. 446–452, 2003**.**

[34] N. Noury, *et al*., “A proposal for the classification and evaluation of fall detectors,” *IRBM*, vol.29, pp. 340–349, 2008.

[35] R. Wilcox “Modern Statistics for the Social and Behavioral Sciences: A Practical Introduction”, CRC Press, 2011.

[36] M. Altaf and J. Yoo, “A 1.83μJ/Classification, 8-Channel Patient-Specific Epileptic Seizure Classification SoC using Non-Linear Support Vector Machine,” *IEEE Trans. Biomed. Circ. Syst. (TBioCAS)*, vol. 10, no. 1, pp. 49-60, Feb. 2016.

[37] V. Franc and V. Hlavac, Statistical Pattern Recognition Toolbox for Matlab, Czech Technical University, 2004, pp. 27–33.

[38] Matlab Statistics and Machine Learning Toolbox, [Online] Available: https://www.mathworks.com/help/stats/

[39] Virtex-5 FPGA User Guide, [Online] Available: https://www.xilinx.com/support/documentation/user_guides/ug190.pdf