Recognizing Falls, Daily Activities, and Health Monitoring by Smart Devices

Sittichai Sukreep,[1*] Khalid Elgazzar,[2] Cheehung Henry Chu,[2]
Chakarida Nukoolkit,[1] and Pornchai Mongkolnam[1]

[1]School of Information Technology, King Mongkut's University of Technology Thonburi,
Thung Khru, Bangkok 10140, Thailand
[2]School of Computing & Informatics, University of Louisiana at Lafayette,
Lafayette, Louisiana 70504, USA

One of the biggest challenges in ageing societies is to improve life, health, safety, and support of the elderly population in their daily life. Currently, the number of elderly people living alone is increasing every year. Living alone allows more freedom but raises the risk of serious injuries or fatal accidents. Falls are the key cause of significant health problems, especially for an elderly person who lives alone. Moreover, vital signs such as heart rate, balancing activities, and environmental context are crucial in relation to the user's condition. To assist people living alone and improve their health quality, we firmly believe that the advances in Smart Devices, Smart Environment, and Internet of Things paradigms are very helpful for developing a fall and activity recognition system. We propose a system using an unobtrusive device consisting of a smartwatch and a smartphone to identify falls and thirteen daily activities (e.g., walking, running, typing, and waving the hand). The events leading to a fall, the speed of falling down, the heart rate while doing an activity, and the time passed since the fall are important data that we store to help a doctor diagnose and rehabilitate a patient. Environment sensors are used to indicate the indices of ambient conditions such as temperature, humidity, brightness, and motion detected. Suitable machine learning techniques are used for daily activity recognition, and the processing time for classification was compared on the basis of a smartwatch and an Amazon Web Services (AWS) cloud server. Threshold-based health risk analysis models are utilized for abnormal activity recognition and heart rate and heat index (temperature and humidity) determination. The system issues different types of notifications such as warning messages, sounding alarms, and phone calls to related persons such as family members, caregivers, or doctors. Various easy-to-understand visualizations are presented to track and monitor the subjects in real time, including heart rate, daily activity summary, health risk status, and environmental information.

---

1. Introduction

Currently, many countries are facing ageing societies.[1] Home safety is a concern not only for elderly people but also for people of any age. It becomes a priority for the elderly, especially if they are living alone. Falls are a primary cause of significant health problems and a leading cause of accidental injuries and deaths.[2] In 2010, more than 662000 adults who suffered from nonfatal fall injuries were hospitalized.[3] If assistance is not swiftly provided, the injuries could be fatal. Timely care and assistance can reduce the hospitalization by 26% and the severity of injuries by 80% if caregivers or doctors are immediately notified after a fall occurrence.[4]

It is necessary to ensure that environmental conditions, such as light, temperature, and humidity, are appropriate for the safety of our loved ones and correctly set up to provide immediate alerts when they need help or assistance.

The daily activity recognition system has become a significant research topic over the last decade with the deployment of automatic, reliable, and cost-efficient health monitoring services. Recently, numerous health tracking systems have been implemented extensively, but there has been a lack of robust proofs regarding the convenience of subjects and the cost efficiency of such systems.[5] An example of a critical scenario is as follows.

Mrs. Michele was an elderly person who was living alone. Her phone was ringing but she was not picking up. It was her daughter trying to ask whether she was ready to go for shopping. The daughter had no way of knowing what was going on and started to panic. She frantically drove to her mom's home, worrying that something serious had happened. When she arrived, Michele was lying on the kitchen floor, unconscious.

At the hospital's emergency room, the doctor asked the daughter how hard Mrs. Michele had fallen and how much time passed since then. She could not answer those questions because Mrs. Michele was home alone. If we knew the conditions and leading to a fall, we would be in a much better position to diagnose and rehabilitate Mrs. Michele, the ER doctor said. Mrs. Michele ultimately received the proper treatment and was discharged from the hospital. However, the doctors recommended that she should be continuously monitored for her condition including heart rate and daily life activities. The daughter was desperately looking for a solution that does not disrupt her working schedule yet offers a high level of privacy, freedom, and convenience to her beloved mom. Deploy video cameras all over the house? No, that is too costly and inefficient and does not offer privacy, the daughter thought to herself.

The above scenario embodies a real challenge that exists in our modern life. We need to take good care of our loved ones, but our busy schedule does not always permit it. To address this challenge, in this research, we focused on the development of an affordable yet highly accurate user-friendly remote health monitoring system using unobtrusive devices.

The major contributions of this study are summarized as follows:
(1) a holistic cost-efficient health monitoring system for indoor settings,
(2) a highly accurate daily activity and a fall detection algorithm, and
(3) an easy-to-understand timeline tracking of user activity using visualization techniques.

The remainder of the paper is organized as follows. In Sect. 2, we highlight related work on fall and activity recognition and classification, devices and wearables, and health monitoring

systems. In Sect. 3, we provide the system architecture and explain the details of the proposed mechanisms. In Sect. 4, we describe the experimental setup and offer insights regarding the obtained results. Finally, in Sect. 5, we explain the visualization techniques used for activity representation and draw conclusions and future directions.

2. Related Works

There are various available fall recognition systems using different techniques, devices, and visualizations methods. Several of these methods or systems require a rather sophisticated hardware infrastructure. This drawback motivates us to overcome these shortcomings through more in-depth investigation of the following areas.

2.1 Fall and activity recognition

Activities of Daily Living (ADLs)[6] is a term used by healthcare professionals to refer to the basic self-caring tasks an individual does on a day-to-day basis. The performance of these ADLs is important to determine long-term care and health coverage that should be applied. Table 1 presents an overview of ADLs detected and monitored in a literature review. There are two types of activity recognition: 1) fall and 2) nonfall activities. Nonfall activities can be subcategorized as movement and nonmovement activities.

Our experiment simulated ADLs and falls based on the literature review. More activities were proposed in order to cover real-life situations.

Table 1
Overview of activities detected from literature review.

Ref.	Devices		Fall	Nonfall															
				Nonmovement								Movement						
	Smartwatch	Smartphone		Sitting	Standing	Lying Down	Holding/Reading	Looking at a watch	Holding a phone	Calling on a phone	Putting down a phone	Walking	Jogging	Waving	Jumping	Typing	Traveling with a phone	Others
(7)		✓	✓	✓	✓							✓	✓					
(8)		✓	✓	✓	✓	✓						✓	✓					
(9)		✓	✓	✓	✓													✓
(10)		✓		✓	✓	✓						✓	✓		✓			
(11)	✓			✓	✓				✓								✓	
(12)	✓					✓								✓				
(13)	✓		✓	✓								✓	✓					✓
(14)	✓	✓	✓									✓			✓			
(15)	✓	✓	✓	✓								✓	✓		✓			✓
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	

2.2   Devices and wearables

The most recent work for fall recognition has categorized approaches into three main types depending on the algorithms and sensors used. The techniques were based on a vision device and image processing; an ambient device, e.g., a motion sensor; and a wearable device with a gyroscope and/or other sensors.[16]

Vision-based techniques: Vision- or camera-based systems for fall detection provide very rich personal and environmental information. Such devices are increasingly used owing to their multiple advantages over other sensor-based devices.[17–19] Núñez-Marcos *et al.*[20] proposed a vision-based solution using convolutional neural networks to recognize a person falling down, and optical flow images were used as input to the networks. Ozcan *et al.*[21] developed a fall detection system through the implementation of an embedded smart camera that employed histograms of edge strengths (ESs) and edge orientations (EOs). Mastorakis and Makris[22] presented a novel fall detection system based on Microsoft's Kinect sensor and similar to that described in Ref. 23.

Ambient-based techniques: Ambient-based device sensors are used to identify falls through an infrared sensor, a single microphone, vibration data, or a pressure sensor. Droghini *et al.*[24] proposed fall and nonfall detection systems using an acoustic signal similar to that described in Ref. 25. Guan *et al.*[26] presented a novel method of detecting a fall using a pyroelectric infrared (PIR) sensor network similar to that of Yun and Song.[27] Nadee and Chamnongthai[28] proposed a method of automatically detecting a falling person using an array of ultrasonic waves connected to a field-programmable gate array (FPGA) processor. The ultrasonic sensor should be placed directly towards objects to acquire the highest measurement accuracy.

Wearable-based techniques: Recently, a variety of wearable devices have been made widely available and have become popular on the commercial market and in the active field of research in pervasive computing.[29] Sztyler *et al.*[30] developed a robust wearable-based activity recognition system from seven wearable devices in different body positions, and the wrist was identified as the most suitable device location at which the acceleration pattern for the activity detection could be measured. Pierleoni *et al.*[31] implemented a fall recognition system using a wearable wireless device consisting of a triaxial accelerometer, a gyroscope, and a magnetometer similar to that described in Refs. 32 and 33.

2.3   Health monitoring systems

Currently, the trend in health monitoring systems is moving from hospital-located devices to portable personal devices.[34] Many existing health monitoring systems have inspired our work. Several of them are available to monitor health, activities, and environment by using different devices and applications for coping with an active lifestyle.[35]

Omoogun *et al.*[36] developed a wearable wireless monitoring system to record the temperature and heart pulse and trigger an emergency response to enable interaction between the patient and the doctor. Jalal *et al.*[37] implemented a healthcare monitoring system to track human movements in a scene by using a camera. Kaninde *et al.*[38] developed a system using

temperature and heartbeat sensors to track the patient's vitality and help in saving precious emergency response time. Similarly, Kakria et al.[39] proposed a real-time heart monitoring system using a wearable sensor.

Bertolotti et al.[40] presented a measurement of a trunk movement system to evaluate the motion and balance control using an accelerometer and a gyroscope sensor. George et al.[41] designed and implemented a patient monitoring system to enable a doctor to monitor a patient's health (temperature, heartbeat, and ECG). Lin and Lo[42] developed a mobile healthcare system, which could monitor physiological data through a smartphone. Similarly, Abdullah et al.[43] presented a system for describing the status of health and fitness.

The advances in technology have resulted in an increased number of wristband or smartwatch devices for health monitoring. The processing power of wristbands and smartwatches has increased, and most smartwatches are now furnished with sensors including an accelerometer and sensors for heart rate, ambient light, and so forth.

In previous works, most researchers used different devices for fall recognition such as smartphones, cameras, and other types of embedded devices, which were obtrusive, cumbersome, and difficult to install. This was rather inconvenient for regular users, especially when sleeping or moving outdoors.

For these simple reasons, a smartwatch was chosen for our recognition of falls and daily life activities instead of a smartphone or other devices. Furthermore, our method could recognize falls and activities using the optimal model from different algorithms to achieve a high accuracy. We also performed a health risk analysis in order to provide a warning in the case of an abnormal or high-risk occurrence. Finally, several easy-to-understand, real-time visualizations were presented.

3. System Architecture and Model

3.1 System architecture

In this section, we describe the system architecture, requirements, and functions of the proposed activity recognition system. Figure 1 shows an abstract overview of the activity recognition system.

The system supports three core functions: (1) activity recognition based on a smartwatch and a smartphone using machine learning algorithms, (2) health risk and environment data analyses based on real-time personal and environmental context, and (3) several visualizations of activity recognition to track the user activity/vital signs including heart rate and other sensors.

Various machine learning techniques are used to recognize user activity on the basis of real-time sensor data collected from both a smartwatch that the user wears on the wrist and a smartphone. Ten common key activities can be detected from a smartwatch including sitting, standing, lying down, walking, jogging, typing, holding/reading, looking at a watch, waving, and falling down. The smartphone can be used to detect activities such as holding a phone, traveling with a phone, and calling on and putting down a phone.
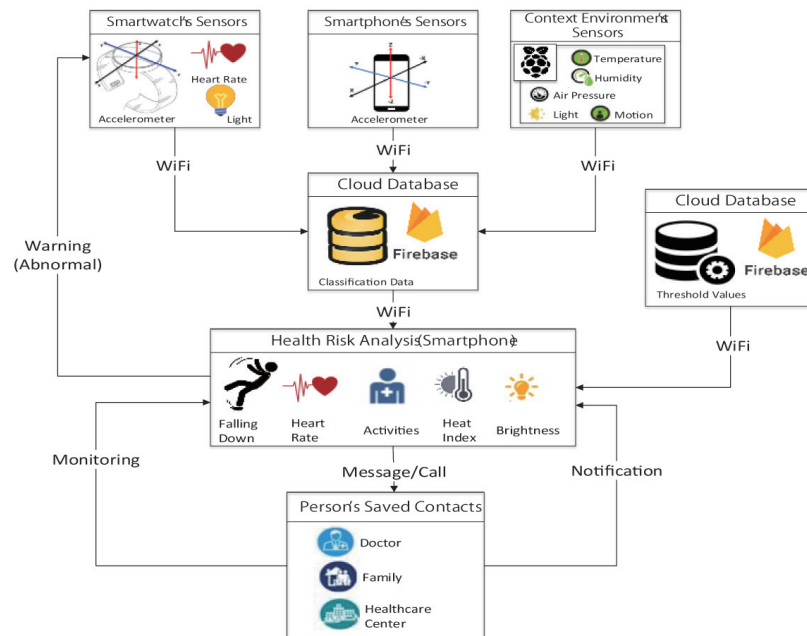
Fig. 1.   (Color online) Abstract architecture of the proposed activity recognition system.

An Android application is developed to gather the data from a smartwatch and a smartphone from a triaxial accelerometer sensor as X, Y, and Z coordinates. The smartwatch is worn on the right wrist first and then the left wrist for training purposes. Activity recognition is performed in two phases: training and real-time recognition.

In the training phase, we generate recognition models using learning by demonstration, where a user performs the target activities and the system records relevant accelerometer sensor data from both a smartwatch and a smartphone. After the iterative data collection is complete, preprocessing and cleaning are carried out to prepare the data for analytics. During this phase, we utilize machine learning models in the WEKA tool[44] to build the recognition models using Naïve Bayes, k-nearest neighbor (k-NN), Decision Tree, and Random Forest classifiers. We try multiple classifiers to select the best and highly accurate one for the real-time recognition phase.

In the real-time recognition phase, the selected optimal classifiers are used to classify the ten target activities captured by smartwatch-embedded sensors and four activities captured by the smartphone with the same stream window size. In addition, the system sets the standard thresholds to check the heart rate, light, temperature, humidity, and air pressure. The system displays several easy-to-understand dashboard visualizations to represent the various captured activities with a timeline. The current activity is displayed in addition to the three most recent activities.

The user dashboard is divided into three main sections: current and past personal activities, real-time vital signs (e.g., heart rate), and environmental conditions surrounding the user. With such rich information, the system can intelligently determine when a high-risk situation occurs or an abnormal condition is detected and issue appropriate alerts to either the user to change activities or to the emergency team to provide immediate medical assistance. The user may also configure the system to contact relatives when prespecified conditions occur.

3.2 Model training phase

In this section, we describe the offline training phase. Figure 2 shows the workflow of the proposed activity recognition system including the following two phases: classifier training and testing (real-time recognition).

3.2.1 Sensor data gathering

The system collects accelerometer, heart rate, and light sensor data from the smartwatch, but only accelerometer sensor data from the smartphone. The triaxial accelerometer consists of X, Y, and Z coordinates as shown in Fig. 3.

The smartwatch data for all activities are subsequently collected from both left and right wrists for 10 s at a 0.2 s sampling rate; this process is repeated three times in 30 subjects for both the smartwatch and the smartphone. These data are sufficient for building robust classifiers. To build classifiers, data from 20 subjects are used for model training, while the data from 10 remaining subjects are used for model testing. In general, there are two types of accelerometer: 1) excluding and 2) including earth's gravity type.

In our experiment, a type 2 sensor was used, and the ranges of accelerometer sensor data are shown in Table 2.
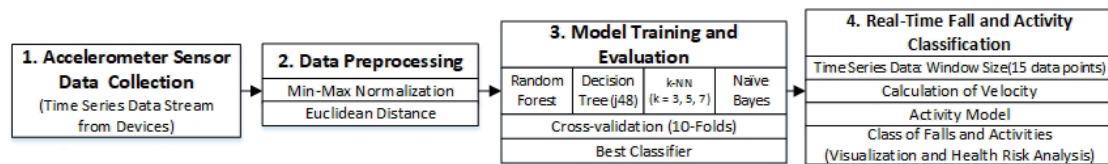
Fig. 2.   Integrated steps of the proposed activity recognition system using smartwatch and smartphone.
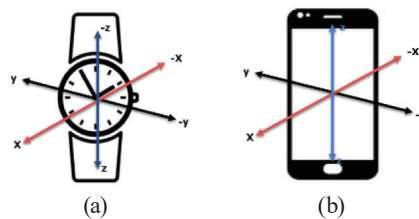
Fig. 3.   (Color online) (a) Triaxial accelerometer of smartwatch. (b) Triaxial accelerometer of smartphone.

Table 2
Accelerometer sensor data ranges.

	Acc_X (g)	Acc_Y (g)	Acc_Z (g)	Diff_X (g)	Diff_Y (g)	Diff_Z (g)	Velocity (m/s^2)
Minimum	−11.8397	−37.8688	−11.1625	−30.9057	−38.1583	−20.7841	0.06103992
Maximum	33.76751	24.29433	20.92284	36.57669	32.74578	19.48476	70.9963731

1. *Acc_X*: *X* coordinate accelerometer sensor data;
2. *Acc_Y*: *Y* coordinate accelerometer sensor data;
3. *Acc_Z*: *Z* coordinate accelerometer sensor data;
4. *Diff_X*: Δ*X* of *X* coordinate accelerometer sensor data between 2 data points;
5. *Diff_Y*: Δ*Y* of *Y* coordinate accelerometer sensor data between 2 data points;
6. *Diff_Z*: Δ*Z* of *Z* coordinate accelerometer sensor data between 2 data points;
7. Velocity: this is a created attribute based on 3 coordinates (Δ*X*, Δ*Y*, and Δ*Z*); the velocity can be used to indicate the activity type;
8. Minimum: minimum value indicates the lowest individual sensor value in each attribute;
9. Maximum: maximum value indicates the highest individual sensor value in each attribute.

We selected four classification algorithms, namely, Decision Tree,[45] Naïve Bayes,[46] *k*-NN[47] (with *k* = 3, 5, and 7), and Random Forest[48] in the training phase using a 10-fold cross validation. The performance of the four algorithms is compared, and the best one is chosen for real-time recognition based on accuracy, as shown in Table 3.

3.2.2 Data preprocessing

In machine learning, data preprocessing is a crucial step to ensure data quality and prepare data for the next stage, which is modeling. Major data preprocessing tasks are handling missing values, outliers, and performing data normalization.

In the first step of our data preprocessing, the data were searched for missing data from both the training and testing sets; there was no missing data in our data set. In the second step, we checked all data for outliers, and found none. In the third step, we applied the Min–Max normalization standard technique[49] to transform all numeric attributes in original data into a new range of [−1, 1]. The Min–Max normalization can be seen in Eq. (1) below.

$$x' = 2 \frac{(x - \min x)}{\max x - \min x} - 1 \tag{1}$$

Here, x' represents a new transformed value in the range of [−1, 1]. The two parameters, namely, (1) $\min x$ and (2) $\max x$, represent the minimum and maximum values of attribute x, respectively.

Finally, the Euclidean distance is used to measure the velocity of the accelerometer data (i.e., the rate of change) as per Eq. (2). The velocity measure indicates how different the data points are from each other; such points can be used to differentiate activities.

Table 3
Results of activity testing (%).

		Algorithm					
		Random Forest	Decision Tree (J48)	*k*-NN (*k* = 3)	*k*-NN (*k* = 5)	*k*-NN (*k* = 7)	Naïve Bayes
Activity	Smartwatch's Activity	99.51*	99.08	99.20	99.13	99.03	87.71
	Smartphone's Activity	99.91*	99.84	99.87	99.85	99.83	99.3

$$Difference(i,\, j) = \sqrt{\left(X_j - X_i\right)^2 + \left(Y_j - Y_i\right)^2 + \left(Z_j - Z_i\right)^2} \qquad (2)$$

Here, Difference $(i,\, j)$ represents the accelerometer data from the ith and jth records, and X, Y, and Z represent the three axes.

The Euclidean distance or velocity measure indicates the type of activity being performed by the subject. For example, the velocity data of walking is relatively greater than those of standing, sitting, and lying down. On the other hand, the velocity of walking is relatively lower than that of jogging or waving, as shown in Fig. 4 for a smartwatch, and Fig. 5 also shows relatively lower velocities for other activities from a smartphone.
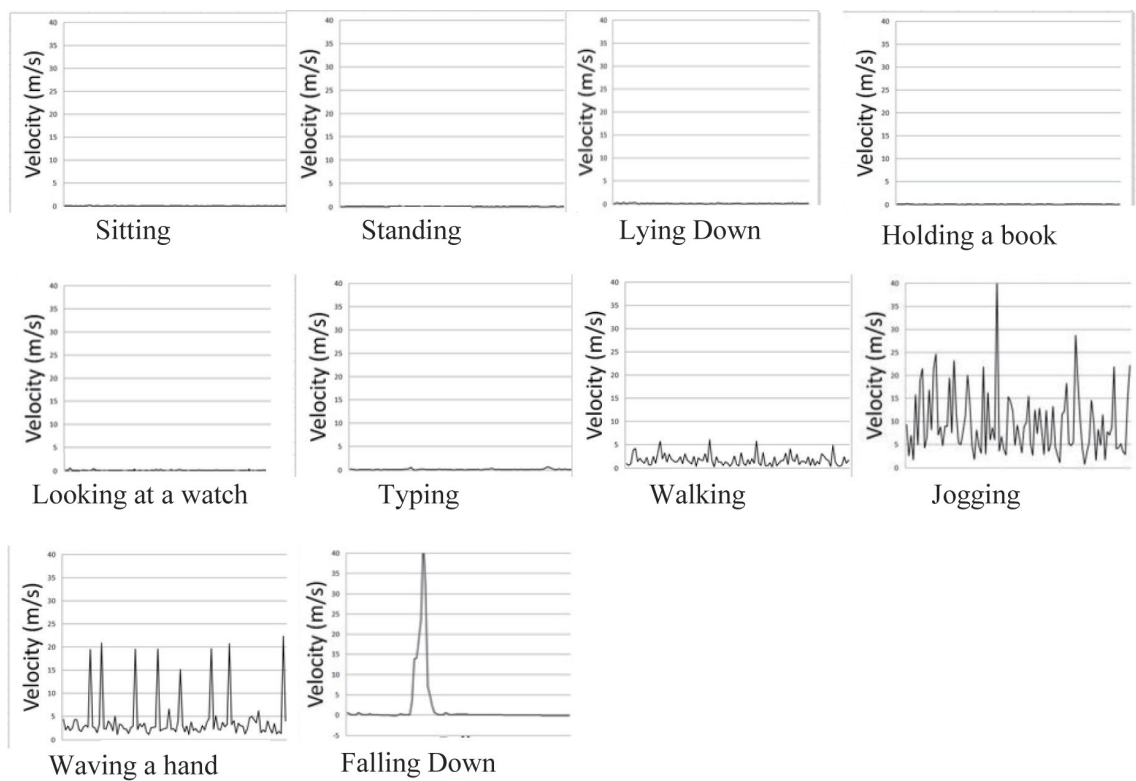
Fig. 4.   Velocity data of different activities from a smartwatch.
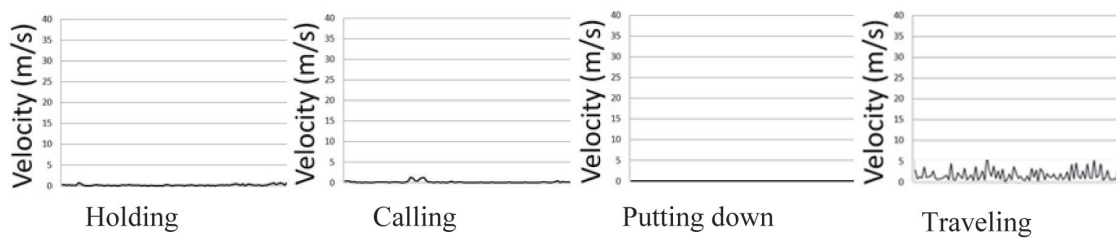
Fig. 5.   Velocity data of different activities from a smartphone.

3.2.3   Machine learning prediction models used

Decision Tree: Decision Tree[45] belongs to the supervised learning algorithms that use a tree to represent a set of decision nodes. Each internal node of the tree corresponds to an attribute, a branch corresponds to a value of an attribute, and each leaf corresponds to a class label.

Naïve Bayes: Naïve Bayes[46] is a classification algorithm based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naïve Bayes uses a similar method to predict the probability of different classes based on various attributes.

k-NN: *k*-NN[47] is among the simplest algorithms for machine learning. *k*-NN is a type of lazy, or instance-based learning, where the function is only approximated locally and all computation is deferred until classification. *k* represents the number of nearest-neighbor data points that we would be looking at in the training data, and we take the most frequently occurring class and assign that class to the new data case.

Random Forest: Random Forest[48] is a supervised classification. It uses an ensemble learning approach by internally building multiple decision trees. The ensemble learning takes advantage of combining voting powers of multiple decision trees to increase robustness and avoids the problem of over-fitting data to our training set. In our model construction, we set two parameters of Random Forest to be 1) seven attributes per decision tree and 2) one hundred decision trees constructed. During testing, the final decision is considered from the majority vote of all decision trees from the forest.

3.2.4   Model training and evaluation

Falls and activity recognition were carried out in the offline phase. Every activity had an equal number of data points. The algorithms chosen to recognize the activity were based on machine learning as shown in Fig. 6.

The data gathered from 20 subjects were contained across ten activities for a smartwatch and four activities for a smartphone. There were a total of 60000 data points ($5 \times 10 \times 3 \times 2 \times 10 \times 20$) from the smartwatch and 12000 data points ($5 \times 10 \times 3 \times 4 \times 20$) from the smartphone. Seven attributes (*Acc_X*, *Acc_Y*, *Acc_Z*, *Diff _X*, *Diff _Y*, *Diff _Z*, and velocity) were selected to build the classifier models.
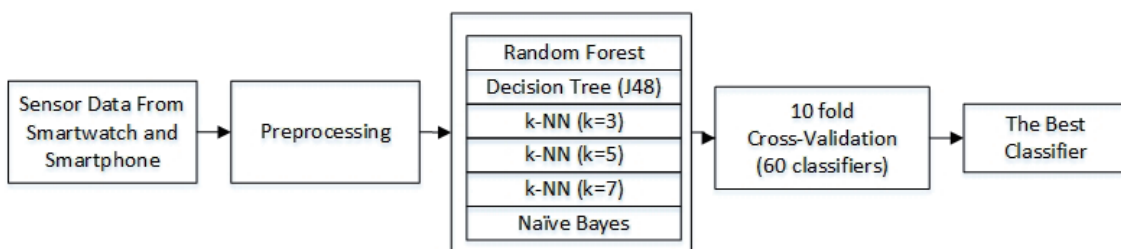

Fig. 6.   Machine learning algorithms used for classification.

To compare the performance characteristics of the various models, it is important to choose the appropriate metrics that measure the algorithm's performance. The results of the classifier are commonly stored in an array known as a confusion matrix. Thus, we chose this matrix to evaluate the different classifiers because it allows for visualizing and learning the performance of the algorithm.

Data from the smartwatch and smartphone were used to evaluate the same algorithms. We used the 10-fold method to cross-validate and classify the activities. Each of the four selected algorithms (with varying $k = 3$, 5, and 7 for k-NN) and a total of ten classifiers made up a total of 60 classifier models, where the best classifier model was chosen for use in the real-time phase.

Table 3 shows the accuracies of the four algorithms. As we can see from the results, the Random Forest algorithm outperforms the other three algorithms. We attribute this to the classifier created for both datasets. Thus, Random Forest will be used for real-time activity recognition.

3.3   Model testing phase

The Random Forest algorithm is used for recognizing various activities in real time as per our performance evaluation. Figure 7 shows the data flow of the proposed system at runtime. First, time series sensor data are collected in the window size. Second, the velocity of the accelerometer data is calculated for each data point in that window. Third, the data are fed to the classifier for recognition. Lastly, the resultant activity is evaluated according to personal health risk thresholds related to the recognized activity (e.g., whether running represents a health concern given the heart rate readings). In such a case, the system will either alert the user or take an automated predefined action such as calling a registered relative or an emergency team.

3.3.1   Data gathering

The triaxial accelerometer sensor data are gathered from ten human subjects for 10 s for each target activity. Each activity is repeated three times for statistical significance. The total number of gathered data points is 30000 ($5 \times 10 \times 3 \times 2 \times 10 \times 10$) from a smartwatch and 6000 ($5 \times 10 \times 3 \times 4 \times 10$) from a smartphone. We set a window size of 3 s buffer time (15 points) to last for one complete data collection cycle, as shown in Table 4.
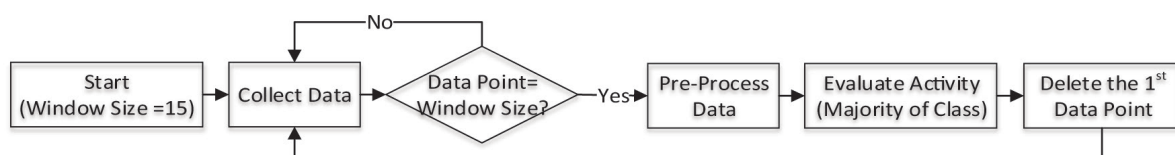

Fig. 7.   Real-time classification procedure.

Table 4
Example of accelerometer data of 3 s buffer time (15 points).

Index	1	2	3	4	5	6	7	8	9	10
Acc_X	−0.097	0.029	0.029	−0.124	0.149	−0.258	0.034	−0.097	−0.112	−0.129
Acc_Y	3.578	3.408	3.772	3.001	3.065	3.046	3.001	3.578	2.680	2.869
Acc_Z	8.664	8.793	8.140	9.018	8.891	9.078	8.851	8.664	9.038	9.090

Index	11	12	13	14	15
Acc_X	−0.107	−0.203	−0.198	0.111	−0.023
Acc_Y	2.761	2.764	2.783	3.061	3.010
Acc_Z	8.985	8.992	9.167	8.626	8.954

3.3.2 Runtime data preprocessing

Similarly to the training phase, data preprocessing is performed at runtime during the recognition phase to ensure the high quality of the collected data and recognition process.

4. Experiment Results and Discussion

In this section, we describe our experimental tests and setup, and provide additional insights regarding the overall performance of the proposed system.

4.1 Experiment setup

A Moto 360 first-generation smartwatch is used for our experiment. It is equipped with a TI OMAP 3 processor, 4 GB internal storage, 512 MB RAM, and Bluetooth 4.0 Low Energy connectivity. The smartwatch runs the Android Wear operating system.[50] A Samsung S5 smartphone is connected to the smartwatch to track subject activity.

The subject tried the smartwatch on both right and left wrists as shown in Fig. 8, and no significant difference was observed in the measurements. The smartwatch reports accelerometer readings, heart rate, and light intensity. The smartphone is used to gather the accelerometer sensor data only.

A set of sensors are used to measure environmental parameters, using temperature, humidity, air pressure,[51] light,[52] ultrasonic,[53] and motion sensors as shown in Fig. 9. In real-life deployments, such sensors can be placed around the areas commonly occupied by the user, such as the living room, bedroom, kitchen, and garage.

4.2 Real-time classification

The results of a classifier are commonly stored in an array known as a confusion matrix. This allows the visualization of the learning algorithm's performance in a specific table. An example of a confusion matrix is depicted in Fig. 10.

True Positives (TP): Number of positive instances that were classified as positive.

True Negatives (TN): Number of negative instances that were classified as negative.
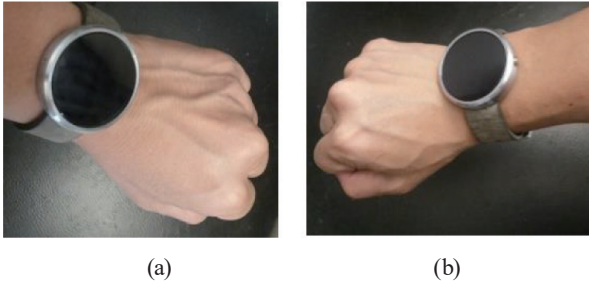
(a)   (b)

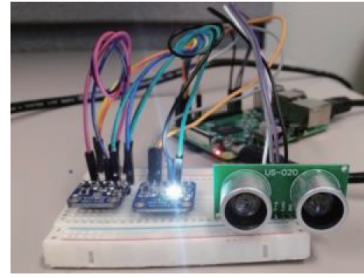Fig. 8.   (Color online) (a) Smartwatch on left wrist. (b) Smartwatch on right wrist.

Fig. 9.   (Color online) Environment sensor devices.

Fig. 10.   Confusion matrix.

False Positives (FP): Number of negative instances that were classified as positive.
False Negatives (FN): Number of positive instances that were classified as negative.

$$accuracy = \frac{TN + TP}{(TP + TN + FP + FN)} \tag{3}$$

The accuracy of the system is the most extensively used performance indicator in classification problems.

$$precision = \frac{TP}{(TP + FP)} \tag{4}$$

The precision or positive predicted value is the ratio of the number of correctly classified positive instances to the entire set of instances classified as positives.

$$recall = \frac{TP}{(TP + FN)} \tag{5}$$

The recall or sensitivity or true positive rate is the ratio of the number of correctly classified positive instances over the entire set of positive instances.

Raw sensor data collected from both the smartwatch and the smartphone are preprocessed at runtime to clean and prepare the data for classification using the Random Forest algorithm. The model's performance is evaluated using Eqs. (3)–(5). The classification algorithm yields average accuracies of 99.19% for smartwatch-based activity recognition, as shown in Table 5, and 100% for smartphone-based activity recognition, as shown in Table 6.

It could be seen that some activities from the smartwatch were misclassified since some values from particular activities were strikingly close to the other activities. For instance, the velocity of waving a hand might be the same as that of walking or jogging, and thus the result of classification could misclassify the activity, as shown in Table 5.

Classification performance is very significant for the real-time system. In this paper, we also consider the processing time to classify an activity on the basis of two different environments: (1) on the smartwatch and (2) on the Amazon Web Services (AWS) cloud server.

On the smartwatch, the classifier is installed directly on the smartwatch in order to classify the activity after gathering the sensor data from the smartwatch. The prediction result will be sent to a cloud database afterward.

Another option is that the classifier is installed on the AWS cloud server. The raw sensor data are obtained from the smartwatch and sent to the cloud database without classification. After that, the data are retrieved from the cloud database using a background application installed on the AWS cloud server for classification. The prediction result will be sent to the cloud database afterward.

The performance is measured by fetching the time-series data stream into the classifier using the window size. One window size contains the data point between 15 (3 s) and 50 (10 s). The data are fetched from one hundred windows one-by-one into the classifier, and this process is repeated in 10 rounds before obtaining the average processing time.

Figure 11 shows the comparison of performance for classification between two environments. The time of performance slightly increases for both environments when the number of data

Table 5
Evaluation results of real-time smartwatch's activities.

Activities/ Evaluation	Sitting	Standing	Lying Down	Holding/ Reading	Typing	Looking at a watch	Walking	Jogging	Waving	Falling	Average
Accuracy (%)	100.0	100.0	96.32	100.0	96.32	100.0	100.0	100.0	99.30	100.0	99.19
Precision	1.000	1.000	0.963	1.000	0.963	1.000	1.00	1.000	0.930	1.000	0.985
Recall	1.000	1.000	0.963	1.000	0.963	1.000	1.000	0.935	1.000	1.000	0.986

Table 6
Evaluation results of real-time smartphone's activities.

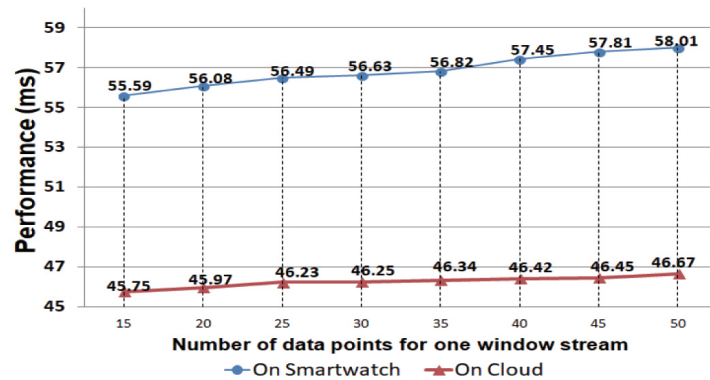| Activities/ Evaluation	Holding	Putting down	Calling	Traveling	Average
Accuracy (%)	100.00	100.00	100.00	100.00	100.00
Precision	1.000	1.000	1.000	1.000	1.000
Recall	1.000	1.000	1.000	1.000	1.000

Fig. 11.   (Color online) Comparison of the processing time's performances.

points is higher. Furthermore, the performance on the AWS cloud server is significantly faster than that on the smartwatch at around 10 ms.

4.3   Activity visualizations

The mobile application provides a rich interactive graphical interface to the user showing the current activity, real-time heart rate, the three most recent activities detected by the system, and surrounding conditions. The application uses simple representative symbols to display such activities. The application also provides a communication platform to interact with the user and healthcare support system.

4.3.1   User dashboard

Figure 12 shows the end user dashboard divided into primarily two sections. The top section shows current and recent activities and vital signs. The bottom section displays the environment context around the user including light, temperature, humidity, and air pressure. Such sensor data information could be relevant or even affect the user's health conditions positively or negatively.

4.3.1.1   Fall and activity recognition

Figure 13 shows the current activity and time elapsed during this activity and the visual representation of the associated health risks. These activities are detected from the smartwatch sensor data analytics.

The system also provides the brightness level around the user as measured by the smartwatch to help users adjust the brightness to a healthy level recommended for a current activity (e.g., reading) to maintain healthy eyesight.[54] An alert message will pop up in case the brightness level[55] is below a certain threshold to be safe for the current activity as shown in Fig. 14.
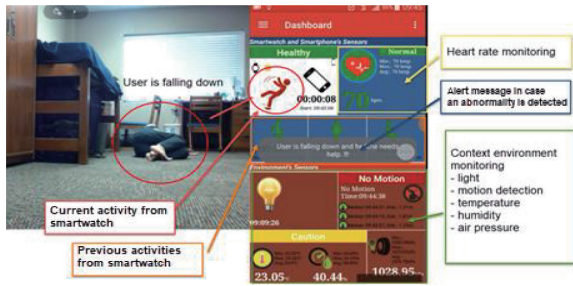
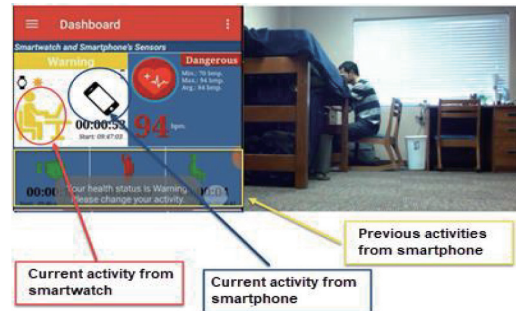Fig. 12. (Color online) Dashboard of activity and environment monitoring.

Fig. 13. (Color online) Current and previous activities from smartwatch and smartphone.
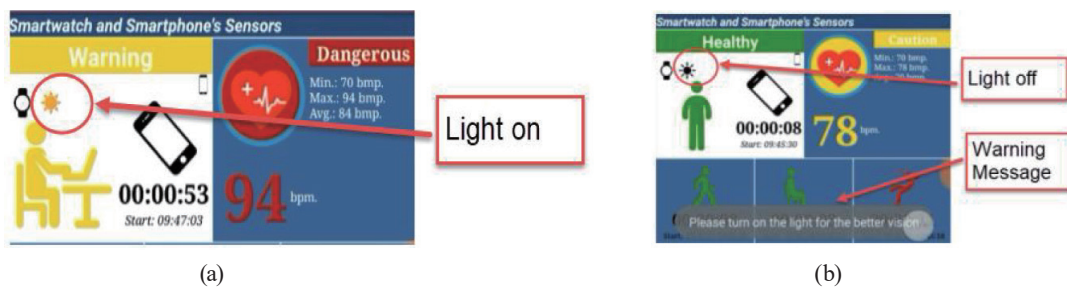

(a)

(b)

Fig. 14. (Color online) (a) Light on and (b) light off and warning message for turning on the light.

4.3.1.2 Environment context

The bottom part of the dashboard shows the environmental context surrounding the user including light, temperature, humidity, and air pressure. This helps the user avoid uncomfortable conditions. The system provides the functionality to identify the level of health risk based on the thresholds of temperature and humidity[56] as shown in Fig. 15.

The iconic colors for the temperature and humidity show different heat indices, ranging from the normal level (green color) to the caution level (yellow color) and the dangerous level (red color).

Additionally, our system provides motion detection functionality using an ultrasonic sensor. The system performs analytics on the basis of raw sensor data from the motion detector to provide rich information including the time and distance between the object and the sensor as shown in Fig. 16.

4.3.2 Health risk analysis

The system performs background analytics on the basis of the raw sensor data collected from personal and environment sensors to provide alerts and recommendations related to the health conditions of the user. For each recognized activity, the system maintains four thresholds that refer to the following health risk levels: normal, low, moderate, and high. The system may correlate between different activities as well as environment conditions to determine the health
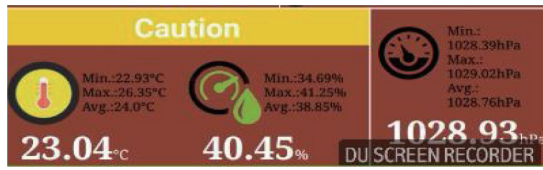
Fig. 15.   (Color online) Environment monitoring and warning system.

Fig. 16.   (Color online) Motion detection system: (a) motion detected and (b) no motion.
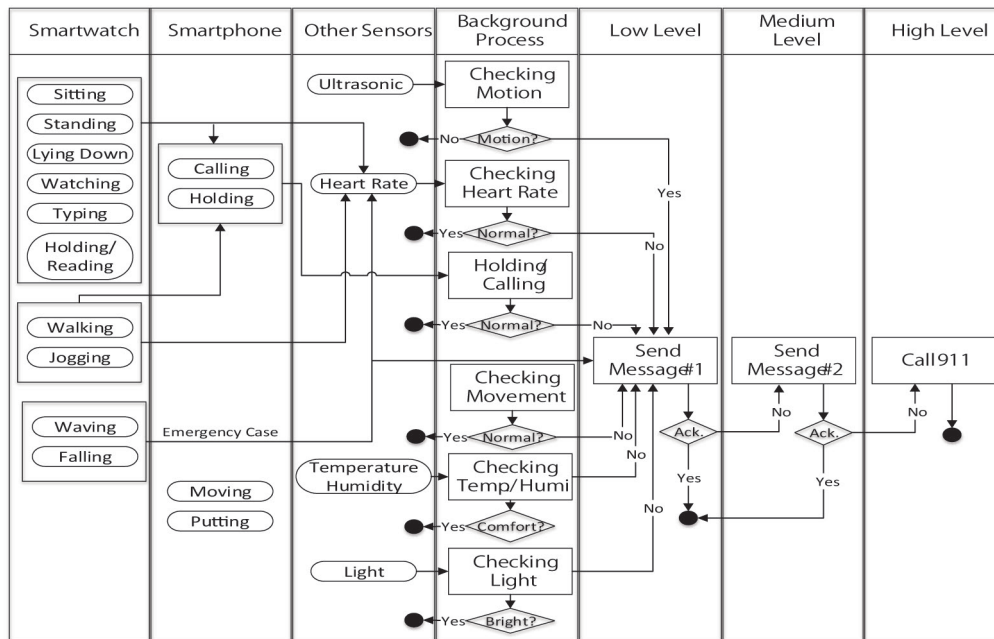

Fig. 17.   Warning system flow.

risk level and issue the appropriate alert. The health risk analysis is shown in Fig. 17. The different risk levels are as follows:

Normal Level: Measurements of vital signs relevant to recognized activities are within the normal range and do not pose any health threats.

Low Level: The vital signs and/or environment conditions exceed the first threshold level, and hence potentially pose a low-level risk to the user's health. In such a case, an alert message will be sent to the smartwatch and smartphone in order to advise the subject to change his/her activity.

Moderate Level: The analysis shows a moderate health risk level when the measured or calculated sensor values exceed the second threshold level. Moreover, in this case, an alert message will be sent to the smartwatch and smartphone recommending a change of activity.

High Level: When the health risk level indicator exceeds the third threshold level, an alert message will pop up and a sound will be emitted to alert the user. If no acknowledgment/ response is received from the user, the system makes a call to a relative or an emergency number.

The color for the activity symbol reflects the health risk level, ranging from normal to moderate to high, as shown in Fig. 18. Additionally, the color of the heart rate follows the same concept.

4.3.3   Activity reporting

The system offers a visualized activity report from historical data using various visualization techniques including pie charts, graphs, bars, and so forth. The reports provide an interactive visual understanding of the user's past activities including health risk levels and percentages.

Figure 19 shows two sample reports, one for the activity summary [Fig. 19(a)] and the other for the heart rate [Fig. 19(b)]. The user can search by date(s), time period, activity type, vital sign, and percentage. The user can also click on a slice of the pie chart on the activity to show more specific details.

Figure 19(b) shows a pie chart at the bottom that represents the summary heart rate risk level. The heart rate status is indicated[57] by different colors: green means a healthy level, yellow represents a caution level, orange indicates an unhealthy level, and red means a risky level.

The system also provides the real-time monitoring of the heart rate during the current activity as shown in Fig. 20. It can show the changes in the heart rate of the subject while
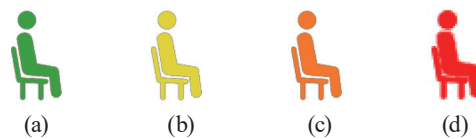

Fig. 18.   (Color online) Health risk levels: (a) normal (green), (b) low (yellow), (c) middle (orange), and (d) high (red).
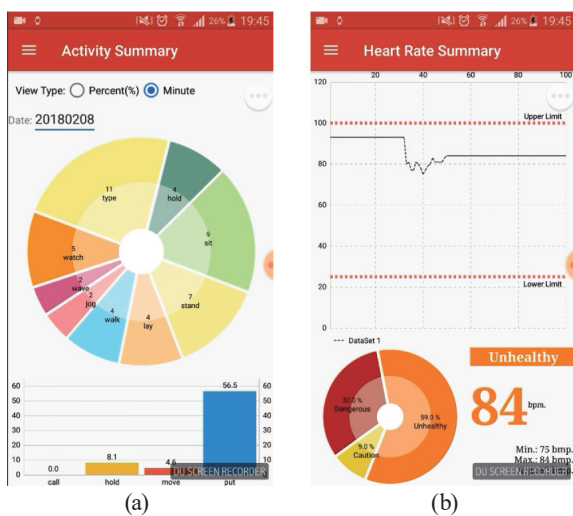

Fig. 19.   (Color online) (a) Summary activity report by date. (b) Summary of heart rate.
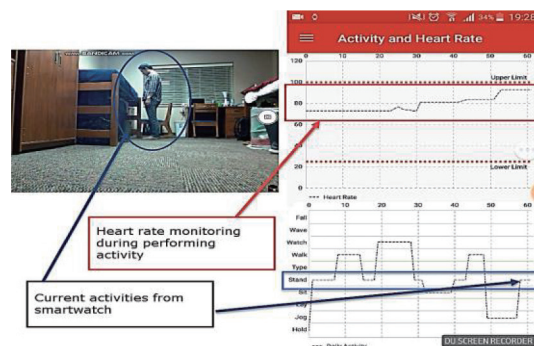
Fig. 20.   (Color online) Heart rate monitoring during activities.

performing the activity. Moreover, the system provides a recommendation of upper and lower limit healthy thresholds for the heart rate in order to alert users in case the heart rate goes over the limit. The thresholds are prespecified per person, and the system can learn over time to adjust these thresholds.

4.3.4 Min–Max activity recognition

It is always helpful for users to know their limits for various activities. The system provides radar graphs with healthy borderlines to track and monitor health risk levels for each of the seven activities (sitting, standing, lying down, walking, jogging, typing, and holding/reading).

The system provides four radar graphs that show recommended minimum and maximum values of the total daily minutes (blue border) for each activity. Health status is color-coded as shown in Fig. 21. The color for the radar graph represents the health status: healthy (green color), as shown in Fig. 21(a), and unhealthy (red color), as shown in Fig. 21(b).

Thresholds typically depend on the individual and must be set by healthcare professionals according to health status, habits, and personal fitness level. One individual may need to be sedentary owing to recovery from a recent surgical operation, while another individual may need to adopt a more active lifestyle.

The system maintains historical data to help diagnose future conditions/incidents (e.g., falling) and assist health care providers to make data-driven decisions. For example, historical heart rate data are of paramount importance to evaluate the patient's health risk while performing activities as shown in Fig. 21(b).

In the event of falling down, the SOS button will be shown immediately on the smartwatch screen, and the user can press the button to request for emergency assistance. The user may choose to wave a hand instead. In both cases, the system automatically places an emergency call within 10 s of the user falling and provides the user's current location.

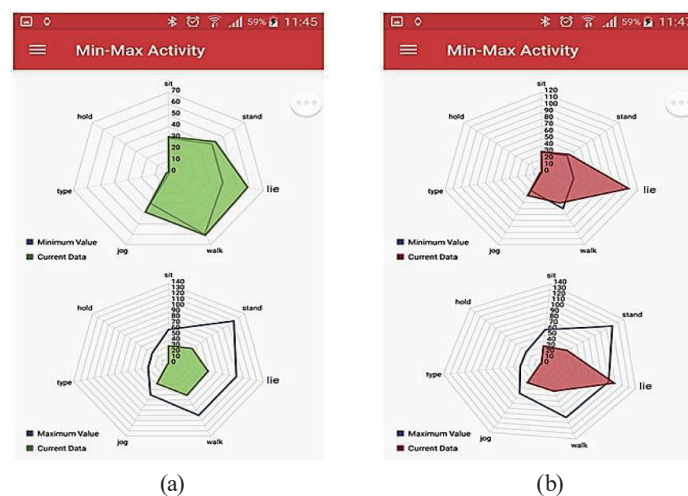(a)   (b)

Fig. 21.   (Color online) Radar graphs of recommended daily activities (minimum and maximum values): (a) healthy and (b) unhealthy.

5.　Conclusion and Future Work

In this paper, we propose a practical, affordable fall and daily activity recognition system using a smartwatch and a smartphone in order to track and monitor the indoor activities of a user. The system provides an unintrusive and convenient way to serve elderly users.

Machine learning techniques were effectively applied to create the best activity classification models for a smartwatch and a smartphone. The data were gathered from a smartwatch for ten activities and from a smartphone for four additional activities. The fall and all-activity recognitions achieved an overall accuracy of 99.19%, a precision of 0.985, and a recall of 0.986 for a smartwatch, and an overall accuracy of 100% and a precision and recall of both 1.00 for a smartphone.

The system can provide a real-time feedback via a tracking and monitoring dashboard screen and introduce several visualizations showing the subject's heart rate, activity summary, and transition activity before and after falling down. Furthermore, the system can warn the user when the health risk level or heart rate exceeds preset limits.

This system is intended for people living alone, such as the elderly or outpatients, in order to track and monitor their daily life activities and behaviors so that we could monitor them for their own safety. Since the system continually tracks daily life activities, in the case of high health risk levels or falls, it will send a warning to those who provide care for them to provide timely assistance.

The system also records and provides useful information from smartwatch, smartphone, and environment data to doctors or caregivers such as the events leading to a fall, heart rate, brightness, temperature, and humidity. Our system can assist people who live alone by improving their quality of life, an important consideration in an ageing society.

Acknowledgments

References

1　R. Stepler: Smaller Share of Women Ages 65 and Older Are Living Alone: More Are Living with Spouse or Children (Pew Research Center, Washington, D.C., United States, 2016).
2　Prevention: National Center for Injury Prevention and Control Web-Based Injury Statistics Query and Reporting System (Wisqars): http://www.cdc.gov/homeandre -creational safety/falls/adultfalls.html (accessed October 2018).
3　N. S. Council: Top Causes of Unintentional Injury and Death in Homes and Communities: http://www.nsc.org/learn/safety-knowledge/Pages/safety-at-home.aspx (accessed October 2018).
4　N. Noury, A. Fleury, P. Rumeau, A. Bourke, G. Laighin, V. Rialle, and J. Lundy: Engineering in Medicine and Biology Society, 2007. 29th Ann. Int. Conf. IEEE (IEEE, 2007) 1663. https://doi.org/10.1109/IEMBS.2007.4352627

5   J. Barlow, D. Singh, S. Bayer, and R. Curry: J. Telemed. Telecare **13** (2007) 172. https://doi. org/10.1258/135763307780908058

6   N. Norbert, A. Fleury, P. Rumeau, A. K. Bourke, G. O. Laighin, V. Rialle, and J. E. Lundy: 29th Ann. Int. Conf. IEEE Engineering in Medicine and Biology Society (IEEE, 2007) 1663–1666.

7   N. Shibuya, B. T. Nukala, A. Rodriguez, J. Tsay, T. Q. Nguyen, S. Zupancic, and D. Y. Lie: Eighth Int. Conf. Mobile Computing and Ubiquitous Networking (ICMU) (2015) 66. https://10.1109/ICMU.2015.7061032

8   S. Sukreep, P. Mongkolnam, and C. Nukoolkit: Recent Advances in Information and Communication Technology 2015 (Springer, Bangkok, Thailand, 2015) p. 215. https://doi.org/10.1007/978-3-319-19024-2_22

9   L. Palmerini, F. Bagalà, A. Zanetti, J. Klenk, C. Becker, and A. Cappello: Sensors **15** (2015) 11575. https://doi. org/10.3390/s150511575

10   Q. Guan, C. Li, X. Guo, and B. Shen: Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2017 9th Int. Conf. (Hangzhou, China, 2017) 373. https://10.1109/IHMSC.2017.91

11   C. Lucio, G. Foresi, A. Monteriù, D. P. Pagnotta, L. Romeo, L. Spalazzi, and A. D. Cesare: IEEE Int. Conf. Consumer Electronics (IEEE, 2018) 1.

12   N. Anne, P. Tseng, M. Paliwal, C. Carpenter, and W. Stipe: Open J. Internet of Things **4** (2018) 87.

13   M. Taylor, M. Canby, V. Metsis, A. Ngu, and C. Rivera: Sensors **18** (2018) 3363. https://doi.org/10.3390/ s18103363

14   B. Serkan, E. A. Sağbaş, and M. Peker: Mobile Solutions and Their Usefulness in Everyday Life, S. Maria, Ed. (Springer, 2019) p. 275.

15   R. Pablo, A. Vega, C. D. Marziani, J. I. Gallardo, J. Pires, and R. Alcoleas: 23rd Congreso Argentino de Ciencias de la Computación (CACIC, La Plata, 2017).

16   M. Mubashir, L. Shao, and L. Seed: Neurocomputing **100** (2013) 144. https://doi.org/10.1016/ j.neucom.2011.09.037

17   K. Ozcan and S. Velipasalar: IEEE Embedded Syst. Lett. **8** (2016) 6. https://doi.org/10.1109/ JETCAS.2013.2256832

18   M. N. H. Mohd, M. Kashima, K. Sato, and M. Watanabe: Int. J. Affective Eng. **14** (2015) 9. https://doi. org/10.5057/ijae.14.9

19   M. N. H. Mohd, M. Kashima, K. Sato, and M. Watanabe: ARPN J. Eng. Appl. Sci. **18** (2015) 8363.

20   A. Núñez-Marcos, G. Azkune, and I. Arganda-Carreras: Wireless Commun. Mobile Comput. **2017** (2017) 1. https://doi.org/10.1155/2017/9474806

21   K. Ozcan, A. Mahabalagiri, and S. Velipasalar: Distributed Embedded Smart Cameras (Springer, New York, 2014) p. 151.

22   G. Mastorakis and D. Makris: J. Real-Time Image Process. **9** (2014) 635. http://doi.org/10.1109/ JBHI.2014.2312180

23   S. Gasparrini, E. Cippitelli, S. Spinsante, and E. Gambi: Sensors **14** (2014) 2756. https://doi.org/10.3390/ s140202756

24   D. Droghini, D. Ferretti, E. Principi, S. Squartini, and F. Piazza: Comput. Intel. Neurosci. **20** (2017) 1. https://doi.org/10.1155/2017/1512670

25   L. Palmerini, F. Bagalà, A. Zanetti, J. Klenk, C. Becker, and A. Cappello: Sensors **15** (2015) 11575. https://doi.org/10.3390/s150511575

26   Q. Guan, C. Li, X. Guo, and B. Shen: 9th Int. Conf. Intelligent Human-Machine Systems and Cybernetics **1** (2017) 373. https://doi.org/10.1109/IHMSC.2017.91

27   J. Yun and M.-H. Song: IEEE Sens. J. **14** (2014) 1482. https://doi.org/10.1109/JSEN.2013.2296601

28   C. Nadee and K. Chamnongthai: Signal and Information Processing Association Ann. Summit and Conf. (2015) 930. https://doi.org/10.1109/APSIPA.2015.7415408

29   H. Martín, A. M. Bernardos, J. Iglesias, and J. R. Casar: Pers. Ubiquitous Comput. **17** (2013) 675. https://doi.org/10.1007/s00779-012-0515-4

30   T. Sztyler, H. Stuckenschmidt, and W. Petrich: Pervasive Mob. Comput. **38** (2017) 281. https://doi.org/10.1016/ j.pmcj.2017.01.008

31   P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti: IEEE Sens. J. **15** (2015) 4544. https://doi.org/10.1109/JSEN.2015.2423562

32   M. Vallejo, C. V. Isaza, and J. D. Lopez: Ann. Int. Conf. IEEE Engineering in Medicine and Biology Society (2013) 1648. https://doi.org/10.1109/EMBC.2013.6609833

33   S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio: Pervasive Mob. Comput. **8** (2012) 883. doi: https://doi.org/10.1016/j.pmcj.2012.08.003

34   F. E. Fajingbesi, R. F. Olanrewaju, B. R. Pampori, S. Khan, and M. Yacoob: Asian J. Pharm. Res. Health Care **9** (2017) 138. https://doi.org/10.18311/ajprhc/2017/14971

35 J. L. Helbostad, B. Vereijken, C. Becker, C. Todd, K. Taraldsen, M. Pijnappels, K. Aminian, and S. Mellone: Sensors **17** (2017) 622. https://doi.org/10.3390/s17030622

36 M. Omoogun, V. Ramsurrun, S. Guness, P. Seeam, X. Bellekens, and A. Seeam: 1st Int. Conf. Next Generation Computing Applications (2017) 169. https://doi.org/10.1109/NEXTCOMP.2017.8016194

37 A. Jalal, S. Kamal, and D. Kim: Int. J. Interact. Multimedia Artif. Intell. **4** (2017) 54. https://doi.org/10.9781/ijimai.2017.447

38 R. Kaninde, A. Gosavi, H. Kariwala, and N. Rathod: Int. J. Recent Innovation Trends Comput. Commun. **4** (2016) 196.

39 P. Kakria, N. Tripathi, and P. Kitipawang: Int. J. Telemed. Appl. **8** (2015) 1. https://doi.org/10.1155/2015/373474

40 G. M. Bertolotti, A. M. Cristiani, P. Colagiorgio, F. Romano, E. Bassani, N. Caramia, and S. Ramat: IEEE Sens. J. **16** (2016) 790. https://doi.org/10.1109/JSEN.2015.2489381

41 M. M. George, N. M. Cyriac, and T. A. Sobin Mathew: J. Res. **2** (2016) 102.

42 Y.-R. Lin and S.-C. Lo: Int. J. Commun. Network Syst. Sci. **9** (2016) 209. https://doi.org/10.4236/ijcns.2016.95020

43 A. Abdullah, A. Ismael, A. Rashid, A. Abou-ElNour, and M. Tarique: Int. J. Comput. Netw. Commun. **7** (2015) 13. https://doi.org/10.5121/ijcnc.2015.7302

44 E. Frank, M. Hall, and L. Trigg: The University of Waikato (2000). http://www.cs.waikato.ac.nz/ml/weka (accessed October 2018).

45 B. Leo: Mach. Learn. **45** (2001) 5. https://doi.org/10.1023/A:101093340

46 J. George and P. Langley: 11th Conf. Uncertainty in Artificial Intelligence (Morgan Kaufmann Publishers Inc., Montréal, Qué, 1995) 338.

47 T. M. Cover and P. E. Hart: IEEE Transac. Info. Theor. **13** (1967) 21.

48 Q. J. Ross: Mach. Learn. **1** (1986) 81.

49 J. Han, J. Pei, and M. Kamber: Data Mining: Concepts and Techniques (Elsevier, Massachusetts, 2011).

50 M. M. Company: Moto 360 (1st Gen): https://motorola-mobility-en-in.custhelp.com/app/home/device/motorola/moto-360-1st-gen (accessed October 2018).

51 Adafruit: Adafruit Bme280 I2c or Spi Temperature Humidity Pressure Sensor: https://www.adafruit.com/product/2652 (accessed October 2018).

52 Adafruit: Rgb Color Sensor with Ir Filter and White Led - Tcs34725: https://www.adafruit.com/product/1334 (accessed October 2018).

53 C. Robotix: Us-020 Ultrasonic Ranging Sensor: http://www.canadarobotix.com/sound-microphone/us020-ultrasonic-ranging-sensor (accessed October 2018).

54 H. H. Publishing: Safeguarding Your Sight: https://www.health.harvard.edu/healthbeat/safeguarding-your-sight (accessed October 2018).

55 E. ToolBox: Illuminance-Recommended Light Level: https://www.engineeringtoolbox.com/light-level-rooms-d_708.html (accessed October 2018).

56 S. Opitz-Stapleton, L. Sabbag, K. Hawley, P. Tran, L. Hoang, and P. H. Nguyen: Clim. Serv. **2** (2016) 41.

57 A. H. Association: Target Heart Rates: http://www.heart.org/HEARTORG/HealthyLiving/PhysicalActivity/Target-Heart-Rates_UCM_434341_Article.jsp (accessed October 2018).

About the Authors

Sittichai Sukreep received his B.S. in computer science from Mahasarakham University, Thailand in 2005 and received his M.S. in software engineering from School of Information Technology (SIT), King Mongkut's University of Technology Thonburi, Thailand in 2013. He is currently a Ph.D. candidate in computer science at SIT. His research interests include data mining, machine learning, and Internet of Things (IoT).

Khalid Elgazzar is an assistant professor and Canada Research Chair with the Faculty of Engineering and Applied Science at University of Ontario Institute of Technology. He is the founder and director of the IoT Research Lab at UOIT. Prior to joining UOIT, he was an assistant professor at the University of Louisiana at Lafayette and an NSERC postdoctoral fellow at Carnegie Mellon School of Computer Science. He received his PhD in Computer Science in 2013 from the School of Computing at Queen's University in Canada. Dr. Elgazzar was a recipient of the outstanding achievement in sponsored research award from UL Lafayette in 2017 and the distinguished research award from Queen's University in 2014. He also received several recognitions and best paper awards at top international venues. His research interests span the areas of mobile and ubiquitous computing, context-aware systems, Internet of Things, and secure and privacy-preserving cyber-physical systems.

Cheehung Henry Chu received his B.S.E. and M.S.E. in computer engineering from the University of Michigan, Ann Arbor, in 1981 and 1982, respectively, and his Ph.D. in electrical and computer engineering from Purdue University in 1988. Since 1988, he has been with the University of Louisiana at Lafayette, where he is currently the Lockheed Martin Professor in the School of Computing and Informatics as well as the Executive Director of the Informatics Research Institute. His research interests are in machine vision, machine learning, and health informatics.

Chakarida Nukoolkit received her Ph.D. degree in computer science from the University of Alabama, USA in 2001. She is now an assistant professor with the School of Information Technology, King Mongkut's University of Technology Thonburi, Thailand. Her current research interests include data mining, visualization, and artificial intelligence.

Pornchai Mongkolnam received his Ph.D. degree in computer science from Arizona State University, USA in 2003 and currently works at the School of Information Technology (SIT) at King Mongkut's University of Technology Thonburi, Thailand. He is the Head of Data Science and Engineering Laboratory (D-Lab) at SIT.