International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 12, No.1 (Jul-2022)

https://dx.doi.org/10.12785/ijcds/120126

Real-Time Wearable-Device Based Activity Recognition Using Machine Learning Methods

Nguyen Thi Thu[1], To-Hieu Dao[2,*], Bo Quoc Bao[1], Duc-Nghia Tran[3], Pham Van Thanh[4] and Duc-Tan Tran[2]

[1]*Hanoi University of Industry, Ha Noi, Viet Nam*
[2]*Faculty of Electrical and Electronic Engineering, Phenikaa University, Ha Noi, Viet Nam*
[3]*Institute of Information Technology, Viet Nam Academy of Science and Technology, Ha Noi, Viet Nam*
[4]*University of Fire Prevention and Fighting, Ha Noi, Viet Nam*

Received 24 Aug. 2021, Revised 8 Feb. 2022, Accepted 9 Mar. 2022, Published 20 Jul. 2022

Abstract: Classification of activities from body-worn accelerometer data to help monitor and take care of health attracts much attention from the research community. This paper proposed to design a real-time monitoring device that can identify people's actions from the accelerometer's data worn on the waist with five activities, including lying, sitting, standing, walking, and jogging. From the collected acceleration data, it is necessary to extract suitable features for real-time classification with high performance. These features are trained with machine learning algorithms that improve the efficiency of action classification. Consequently, a decision tree algorithm was embedded in the microcontroller. This programmed waist-mounted device was connected to the monitoring system via WiFi protocol. Users could monitor activities and managed data on a computer, a website, or a smartphone. The results were optimistic when the overall accuracy for the activities dataset reached 99.3% when training and classifying the activities on the computer. When experimenting with real-time wearable devices, the overall accuracy when classifying activities decreased but was still very good, reaching over 90%.

Keywords: Human activity, classification, machine learning, decision tree, webserver, accelerometer.

1. Introduction

Over the past few decades, research on human activity has grown tremendously and had many life applications. It can be applied in healthcare [1], [2], accident detection for the elderly [3], [4], industry [5], security [6], etc. Activities classification technology collected information about user behavior, provided various ways of interaction, thereby allowing the system to actively support users in their work [7]–[10]. There were three common approaches to classify human activity: i) computer vision-based methods [11]–[13]; ii) advanced technology-based approach with pre-established conditions [14], [15], and iii) methods based on sensors attached directly to the body combined with machine learning algorithms [16]–[21].

Photo or video is used for a computer vision (image/video processing) approach to detect human behavior and changes in the surrounding environment. However, the use of surveillance cameras had many limitations due to the limited image analysis process in low light conditions, high investment costs, and the risk of affecting user privacy. Similarly, the pre-established conditions approach was difficult to apply in places with limited infrastructure because of the need to set the necessary conditions in advance, the investment costs of research, operation, and no minor maintenance.

Different from the above two approaches, the body-mounted sensor-based approach has opened up many potential applications in activity classification [16]–[20], [22]–[30]. This method did not require pre-setting conditions and did not depend on space and time of use. This research focused on exploiting the ability to real-time classify basic human activities using sensors mounted on the human body.

In the era of development with many advanced manufacturing and manufacturing technologies, sensors were becoming more and more diverse, measurement parameters were improved in terms of accuracy, and were applicable in many other problems together. Therefore, from the sensor data collected by attaching the accelerometer sensor to the waist, the authors proposed to use a solution to segment data by time (sliding window) with a fixed width n - seconds. This research built a low-cost and real-time activity classi-

E-mail address (corresponding): hieu.daoto@phenikaa-uni.edu.vn*, nghiatd@ioit.ac.vn, tan.tranduc@phenikaa-uni.edu.vn*

fication device with highly efficient features and a simple classification algorithm with high accuracy. Consequently, the proposed model used the decision tree algorithm [8] with five activities: lying, sitting, standing, walking, and jogging.

2. Research Methods

A. Activities identification process

To build a human activity classification model, the authors proposed to use two features of acceleration data: Mean and Standard Deviation (SD). In this research, the classification model using the decision tree (DT) algorithm is deployed to improve classification performance. The model structure has three main stages: data collection, feature analysis, and activities recognition, as shown in Figure 1.

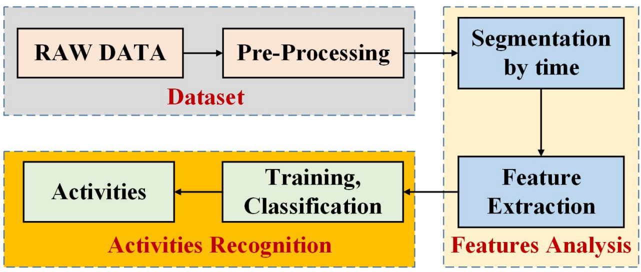

Figure 1. Block diagram of classification state process

The monitoring device was mounted directly on the waist and collected accelerometer data in three X, Y, and Z axes. Before moving to the feature analysis stage, sensor data is processed to eliminate errors (pre-processing). Next, this data was segmented using a fixed-width n-second sliding window of each activity. Then, each feature vector containing the typical property was extracted from the information on each data segment. These features were trained to build classification models and classify actions. One device was integrated into a low-cost accelerometer and a wireless data transmitter/receiver (WIFI/INTERNET) to collect activities data. It was mounted on the waist allowing to transfer the information of the classified actions to the server. Users could easily observe on smartphones or WebServer in real-time. This system significantly reduced user limitations in the data collection and aggregation process. The block diagram of the proposed device is illustrated in Figure 2. The accelerometer was connected to the MCU (Multipoint Control Unit) through the I2C communication standard (Inter-Integrated Circuit) interface to acquire accelerometer data in three axes X, Y, Z. This data was written on the memory card (external memory). In this design, the collector used a 3.7V-850mAh lithium battery.

After embedding the algorithm (Figure 3), the MCU connected the peripherals and set them to the operating threshold values in the initialization loop. At each loop, the MCU sampled the acceleration value in 3 axes X, Y, Z with a frequency of 1HZ and transmitted to the Server via WIFI

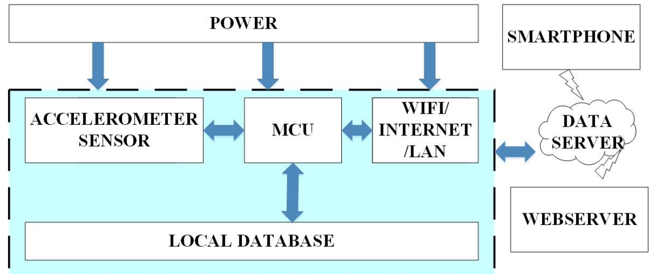

Figure 2. The block diagram of the proposed system

(INTERNET/LAN) according to HTTP (Hypertext Transfer Protocol) and GET method. After 6 seconds, the system would extract features and classify activities.

B. Data collection system

After collection, this data is processed to remove the lost signals and remove noise during transmission. It was labeled in five classes, including activities: lying, sitting, standing, walking, and jogging (Table I).

TABLE I. Definition of human activities

ACTIVITY	DEFINE
Lying	Put whole body in a flat position so that you are not standing or sitting.
Sitting	Rest your body on your bottom with your back straight.
Standing	To be on your feet and in a vertical position.
Walking	Move somewhere by putting one foot in front of the other on the ground, but without running.
Jogging	A form of trotting or running at a slow or leisurely pace.

Data samples were collected with a frequency of 1Hz on all 3 axes X, Y, Z. Therefore, the time taken to collect data can be calculated based on the number of data samples. Besides, based on the time window size, we aimed to build a homogeneous data set for each action. So every six seconds the system stored six data samples and conducted real-time action classification (Figure 3).

The accelerometer ADXL345 incorporated an ADC (10-13 bit) that could measure dynamic and static acceleration [25]. Acceleration was calculated according to equation (1).

$$A_i = \frac{\left(\frac{Sam_i}{1024} * R - O_i\right)}{S_i} \tag{1}$$

A_i: Acceleration value in the direction i (i = X, Y, Z);

Sam_i: Value after sampling axis i;

R: Voltage reference;

O_i: Compensation;

S_i: The sensitivity of the acceleration in the direction i.

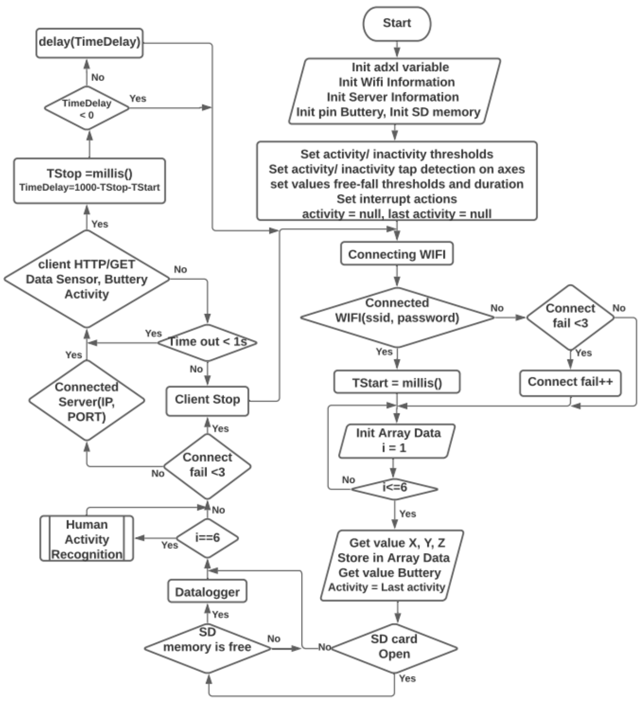

Figure 3. The flowchart for our device

Figure 4. Location of monitoring equipment

Data logging was done within the given time period. The experimental process was conducted on people aged 18-32, 1m43-1m75 in height, with a device attached to the waist area through a heat-resistant belt (Figure 4). These volunteers belong to the scientific research group of the Faculty of Automation Technology. When experiment, they wore the mask and were kept information confidential. The data collection device had the following dimensions: $78 \times 50 \times 30mm^3$, weighs about 140g, the X-axis towards the ground, and the Y-axis to the right. Details were as follows: We had conducted experimental data recording with a group of 15 volunteers (eight man and seven women, age: 18-32, height: 1.56-1.75m, weight: 43-62 kg, selected from the University of Information and Communication Technology). The volunteers held our proposed device on their waists with a sampling frequency of 1Hz. Information collected dataset added include: lying: 4200; sitting: 3432; standing: 3240; walking: 4368; jogging: 3030.

Measured data from activities were segmented to increase recognition performance. Measured data from activities were segmented to improve recognition performance. There were many methods for segmentation, such as segmentation using sliding windows, energy-based segmentation, rest-position word segmentation, time data segmentation [26], and extended data. In this paper, the research team used sliding windows with a fixed size over time. If the window size was too short, the rate of duplication in the collected data increased. However, if it was too long, the probability of multiple activities occurring in a sliding window rose. Therefore, it was necessary to experiment with different window sizes to choose the most optimal value. Kwapisz [26] segmented the data with the 10 seconds (10s) sliding window method and demonstrated that the 20s window size gave less promising results.

TABLE II. Composition of activity observations public data

Activity	Sliding window
Lying	540
Sitting	572
Standing	700
Walking	728
Jogging	505
Total	3045

Therefore, the collected data set was segmented into 3045 windows (Table II) with each window size of 6s. An example of the data segment record sample was shown in Table III. Acceleration value represented in g.

C. Data Acquisition

Features extraction from activities data helped the classification model to achieve high performance. We analyzed two features to determine the range of values and features

TABLE III. Data segment record template

X(g)	**Y**(g)	**Z**(g)	**TIME**
0.14	0.03	1.03	2021-08-03 09:44:30
0.14	0.04	1.02	2021-08-03 09:44:31
0.15	0.03	1.03	2021-08-03 09:44:32
0.14	0.03	1.03	2021-08-03 09:44:33
0.14	0.04	1.02	2021-08-03 09:44:34
0.14	0.04	1.03	2021-08-03 09:44:35

corresponding to each activity obtained, including Mean and Standard Deviation (SD). According to formulas (2) and (3), these features were extracted with the acceleration values on the X-axis. The same to the Y-axis and Z-axis. The unit of acceleration is g. Formulas of acceleration features on the X-axis are given by:

$$Mean: \quad \mu(X_j) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{2}$$

$$SD: \quad \sigma(X_j) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{3}$$

Where:

X_j is the record j;

x_i is the sample ith of record X_j;

N is the total of data samples (a window size);

$\mu(X_j)$ is the mean of X_j.

TABLE IV. Acceleration data features on the X, Y, and Z axes

	Axis	**Lying**	**Sitting**	**Standing**	**Walking**	**Jogging**
Mean	X(g)	-0.064	-0.874	-0.934	-0.928	-0.932
	Y(g)	0.823	0.37	0.085	0.072	0.004
	Z(g)	0.501	-0.105	-0.024	0.03	-0.032
SD	X(g)	0.018	0.011	0.0085	0.184	0.488
	Y(g)	0.022	0.023	0.025	0.206	0.42
	Z(g)	0.021	0.045	0.03	0.111	0.209

The aggregated data in Table IV showed that the Mean features of the three actions of standing, walking, and jogging tended to fluctuate around a central value on the X-axis of -0.93g. Besides, the domain of Mean feature values on Y and Z axes of sitting and standing activities was much larger than that of standing, walking, and jogging. For example, lying and sitting had a minimum value on the Y-axis of 0.37g, while the activities of standing, walking, and jogging had a maximum value on the Y-axis of 0.085g. Not only that, between the lying and sitting, there was a marked difference in feature values. For example, lying had the Mean feature on the Z-axis was 0.501g and for sitting was -0.105g. Therefore, the Mean feature could distinguish whether the lying or sitting activity. At the same time it made a very clear distinction between movable action (walking, jogging) and sedentary action (lying, sitting, standing).

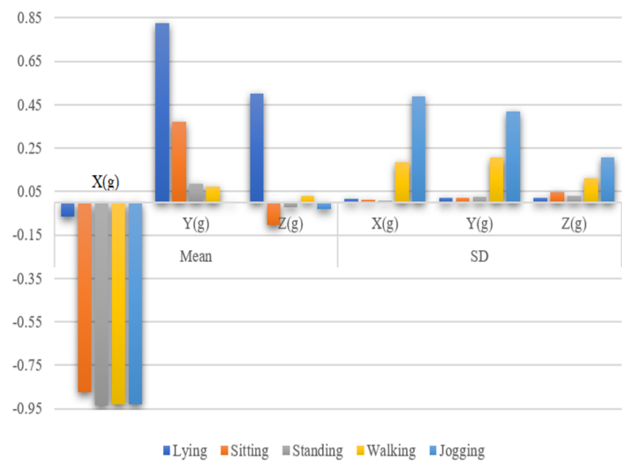

Figure 5. Feature chart on a window

Unlike the Mean feature, the dynamic state (walking, walking fast) had the SD feature values different from the static state (standing, sitting, lying) as Figure 5. For example: feature value of moving person SD_{min} = 0.111g while in non-moving state, feature value SD_{max} = 0.045g. Besides, the SD feature had a clear difference between the two activities of walking and jogging. For example: According to the X-axis, walking had SD= 0.184g compared to jogging with SD = 0.488g. Thus, this feature could be used to distinguish a person in a static or dynamic state, walking or jogging.

D. *Feature extraction*

To reduce the number of data sizes to make data observation more intuitive, the research team used the t-Distributed Stochastic Neighbor Embedding method (t-SNE) [20]. This algorithm would map each feature extracted in the 3-dimensional space (X, Y, Z) to the 2-dimensional space without deviating the feature properties. Figure 6 showed that without using features, duplicate activities were almost indistinguishable.

Three activities of walking (green area), jogging (orange), and standing (pink area) showed overlap. Besides, the two activities, lying (black area) and sitting (blue area), were separate from the other three activities. This gave information that data segmented with windows of 6s has good performance when used to classify human activity.

Using the Mean feature, Figure 7 illustrated walking (green area) and fast walking (orange area) have been separated from the rest. In contrast, lying (black area),

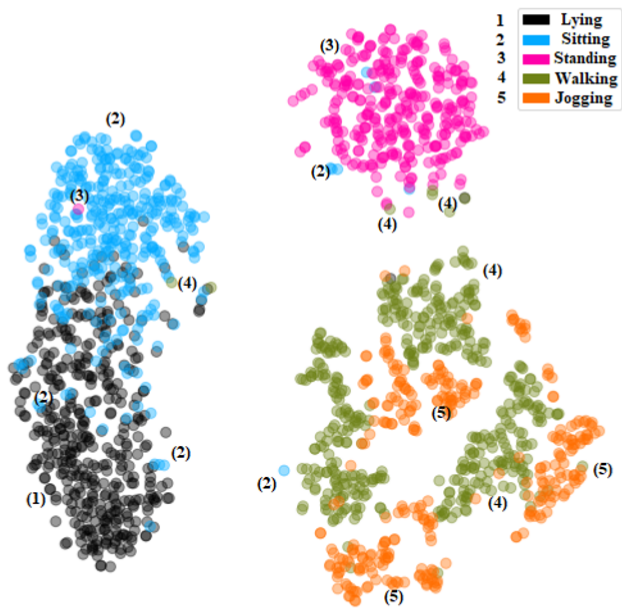

Figure 6. Mapping dataset without using features

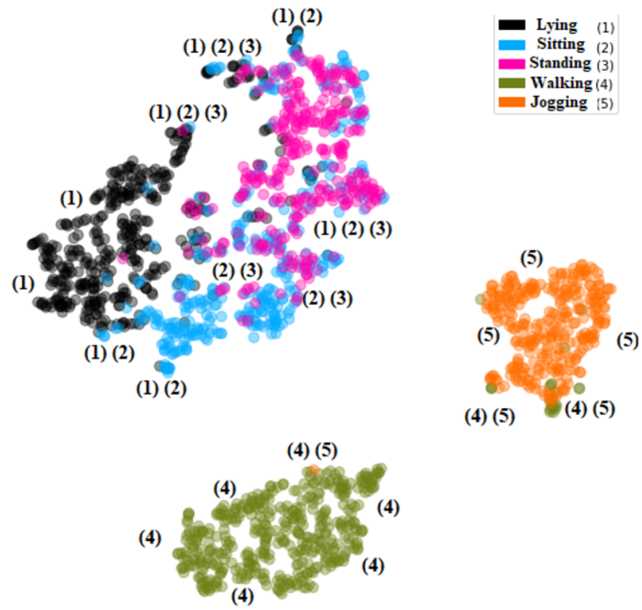

Figure 8. Standard Deviation feature

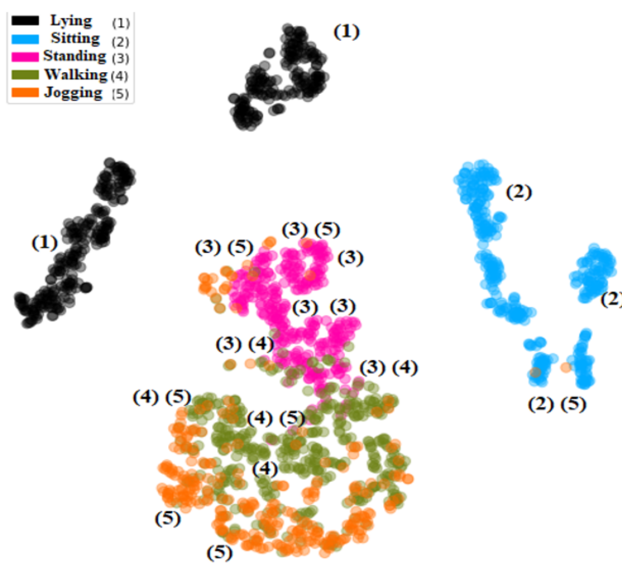

Figure 7. Mean feature

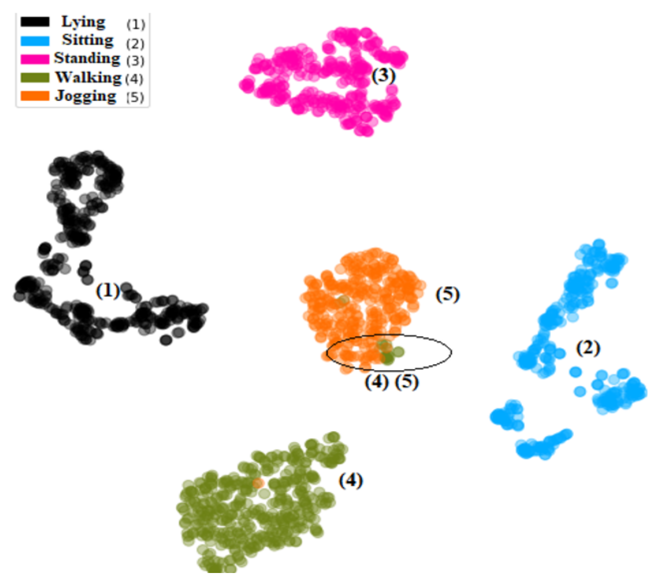

Figure 9. Combining SD and mean features

sitting (blue area), and standing (pink area) had significant overlap. It was almost impossible to classify.

With the SD feature (Figure 8), walking (green area) and jogging (orange area) were no longer confused with each other and better differentiated from the rest. However, sitting (blue area) and standing (pink area) were difficult to distinguish. Besides, lying (black area) was more clearly separated, but it is still confused with the two activities of sitting and standing. It showed that the SD feature was highly effective when classifying walking and fast walking.

With the combination of two features: SD and Mean, the activity classification has achieved good results. Figure 9 illustrated a slight confusion between walking (green area) and fast walking (pink area). However, the boundaries between activities were more clearly defined than when the feature was not used (Figure 6).

E. Classification Algorithm

1) Machine learning for human activity recognition

Classification algorithms were part of supervised learning commonly, and they used to recognize actions on people. From the data collected from the wearable sensor, the classification algorithms trained the data and used it to perform the classification against the new data. Popular classification algorithms included: Decision Tree, Support Vector Machine, K-Nearest Neighbor, Random Forest, Gradient Boosted Decision Tree.

Decision Tree (DT) was a supervised learning model applied in both classification and regression problems. It was a tree-like diagram, where each inner node represented a check on an attribute, each branch represented a test, and each leaf node (terminal node) held a class label. DT was a good exploratory method if the most influential features were selected in the dataset. It learned a series of if-then rules based on thresholds of feature values that led to predicting the target value.

Support Vector Machine (SVM) were supervised learning models with related learning algorithms for data analysis used for classification, regression, and outlier detection. However, it was primarily used for classification targets. SVM took incoming data and classified them into two different classes. Therefore, it was a binary classification algorithm. An SVM model represented the events in terms of points in n-dimensional space (n is the number of features used in the algorithm), with each value of each feature being a part of the association. An SVM model represented points in space and chose the boundary between two categories so that the distance from the practice events to the boundary was as far as possible. New events were also represented in the same space. The algorithm predicted them to fall into one of two categories depending on which side of the instance's boundary.

K-Nearest Neighbor (KNN) was an algorithm to find the output of a new data point by relying only on the K closest data points in the training dataset, regardless of some data points in the nearest data points. KNN was very easy to use and install, the computational complexity was small, and the prediction of new results was relatively easy.

Random Forest (RF) was a popular and effective learning method for classification, regression. RF could be thought of as a set of decision tree algorithms (DTs) at a high level. An RF generated a lot of individual DTs per training set. The idea was that the DT branch in an RF predicted the target values in the training set but might be different from other DT branches in the RF. This distinction was made by introducing random variations into the construction of each DT. These random variations occurred in two ways: I) the data used to generate the DT were random; II) the feature selected in the classification test was randomly selected.

Gradient Boosted Decision Tree (GBDT) like RF, GBDT used a cluster of multiple DTs to create a more robust predictive classification model. Unlike RF that built and combined different random DTs in parallel. The idea of GBDT was to make a series of DTs, where each trained DT tried to correct the errors of the previously trained DT. Once the model was built, the prediction with GBDT was fast and didn't take much memory. The main parameters that determined the complexity of a GBDT model were the number of DTs, the number of training data in the pool, and the learning rate.

In this study, we have applied Decision Tree algorithm for activity recognition and classification device to build a simple embedding algorithm and operate in real-time.

2) *Classification Algorithm used on our device*

After being extracted from the dataset, the decision tree algorithm trained Mean and SD features. Feature windows were split 60% for training and 40% for testing. Based on the training data, the decision tree algorithm was limited to a max depth of 4 and this algorithm would help us to predict the appropriate decision threshold with the used feature values (Figure 10).

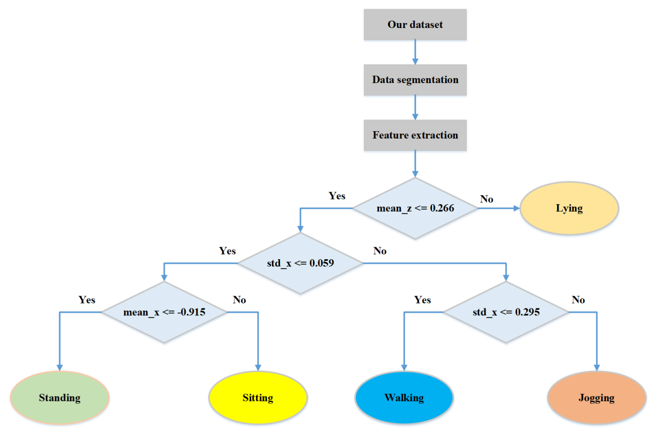

Figure 10. Decision tree for testing

When performing real-time activities classification, MCU needed to meet the memory requirements and complexity of the model. Therefore, the decision tree classification model was suitable for implementing real-time human activities classification. Rules for operating the device must ensure that the device was fixed at the waist and the three axes of the acceleration direction as shown in Figure 4, limiting the position changed too quickly and avoiding strong impact on the device. The collected data was stored on the memory card with the following format: $sequencenumber, x-axisvalue(g), y-axisvalue(g), z-axisvalue(g), samplingfrequency$.

F. *IoT Applications*

Mcu Esp8266

The internet of Things (IoT) is developing rapidly and being applied in many fields [31]. IoT powers the connectivity needed to integrate intelligent machines, components, and products in the ongoing Industry 4.0 trend. WIFI technology (IEEE 802.11) made it possible for devices to connect directly to existing WiFi infrastructure and accessed the Internet with reduced communication latency and low system costs. In this research, our central processor ESP8266 [32], [33], shown in Figure 11b, was integrated with WIFI technology, had low cost, and was energy efficient. It gave information about suitability for IoT applications that use lithium batteries (limited power source), allowing for thousands of charge cycles.

Storage And Communication

The memory on the ESP8266 had 4MB of flash, and at least 3MB can store data, but this memory could only read and write about 10000 times, which was relatively low for a data acquisition system. Therefore, the collected data was stored on external memory (local memory) or memory card and an online database to keep the data intact and not lost during the collection process.

Micro SD memory card used the SPI (Serial Peripheral Interface) standard to transfer data without interruption. Any number of bits could be sent or received in a continuous stream. Collected data stored on the card was saved as a TXT file and was numbered, collected by the timer operating on the ESP8266. The built-in storage was expandable with unlimited capacity.

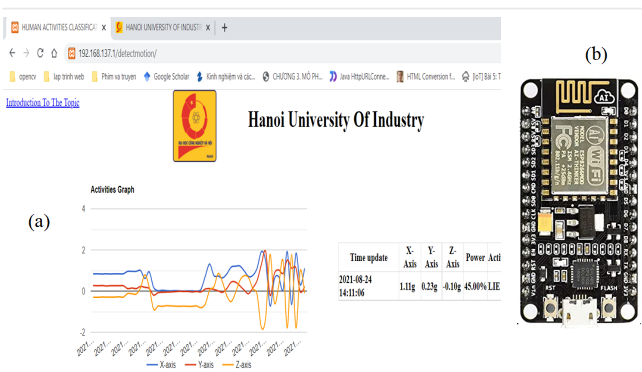

Figure 11. WebServer interface (a) and MCU ESP8266 (b)

Storing data on extended memory made it difficult to observe the collected data and accessed the data. Therefore, ESP8266, after storing data on external memory, would transfer data to the server via a WIFI connection. This controller contained a library that supports HTTP/GET and TCP/IP communications. This paper used the GET method to push data to the server via HTTP protocol with JSON standards. Data would be continuously updated into the database on the server in real-time, including acceleration in X, Y, Z axes, battery capacity (%), and data update time (time automatically updated according to the server's time zone). By building a MySQL database on XAMPP software, users could directly manage the database or access it via the Website (Figure 11a) using the PHP (Hypertext Preprocessor) programming language.

With the Web Server interface, users could monitor the monitoring object's activities in real-time and graph the acceleration change in 3 axes with a 6-second data segment.

Apps On Smartphone

For convenient data observation, we built an activities recognition application on the phone based on the sensors available in a device (Figure 12). The smartphone application interface was built to help users easily observe the parameters collected from the sensor in units of acceleration g or m/s^2 and time to update data from the sensor. Besides, the battery capacity display function helped users know when the device would stop working and charge the battery. Users could click the English/Vietnamese button to switch the display language interface.

With a battery capacity greater than 20%, the battery icon background would be green and vice versa. In the lower frame, the application would display the action classification results corresponding to the activity image.

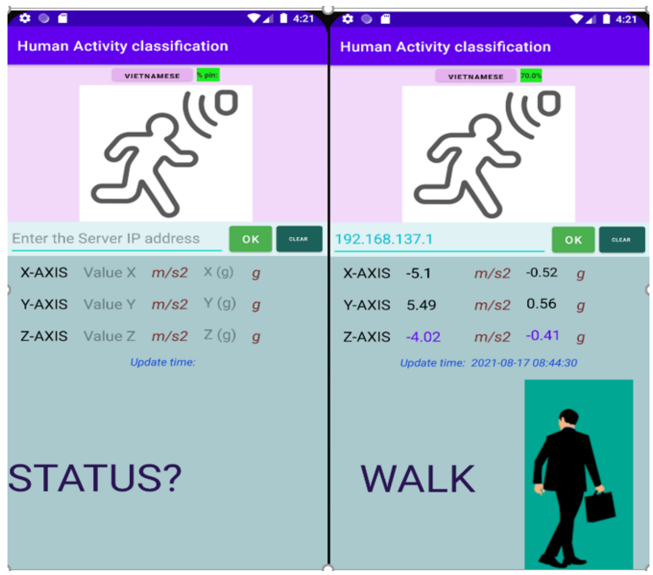

Figure 12. Activity classification apps

To use the action recognition application, users needed to ensure the server is always online with the address (IP) and PORT permanently fixed. This paper used a Dell G515 laptop as a server with a fixed IP/PORT address of **192.168.137.1:80**. The collection device was labeled and guaranteed to access WIFI shared with a Server with the same DNS (Domain Name System) and default gateway address. Users could not connect to data and display if the server IP or PORT address was wrong or the server was not started.

3. Results

A. Evaluation method

The performance of the usage model would be evaluated according to the following three parameters: accuracy (acc), sensitivity (sen), and specificity (spe). The above values are determined based on the Confusion Matrix [23] for each activity:

$$acc = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

$$sen = \frac{TP}{TP + FN} \quad (5)$$

$$spe = \frac{TN}{TN + FP} \quad (6)$$

True positive (TP) was when activity happened, and the device predicted it to happen. False positive (FP) was when an activity didn't happen, but the device predicted it to happen. False negative (FN) was when an action happened, but the device predicted different activities to happen. True negative (TN) was when an activity didn't happen, and the device also predicted that it didn't happen.

B. Evaluation of the classification model

In this paper, we compared the proposed model with those using machine learning algorithms such as Decision Tree (DT), Gradient Boosted Decision Tree (GBDT) [34], [35], Support Vector Machine (SVM) [36], Random Forest (RF) [37], K-Nearest Neighbor (KNN) [38]. The data features were trained and evaluated in Table V.

TABLE V. Comparison when applying some common algorithms

Feature	DT	GBDT	SVM	RF	KNN
Mean	85%	92.7%	91%	93.4%	92%
SD	80.3%	84.6%	88%	84%	83%
Combined	99.3%	99.6%	100%	99.6%	100%

The best performance when using the GBDT classifier with the Mean feature was 92.7%. Yet, the lowest accuracy of the classification models was 85% with the decision tree model (DT) and the SD feature. However, the combination of SD and Mean features resulted in a spike of 99.3% (Figure 13). The detailed results of the decision tree model are presented in Table VI.

TABLE VI. Confusion matrix

Activity	Proposed Activity				
	Lying	Sitting	Standing	Walking	Jogging
Lying	278	0	0	0	0
Sitting	0	228	0	0	0
Standing	0	0	215	1	0
Walking	0	0	2	286	4
Jogging	0	0	0	4	198

The number of correctly classified activities cases and the number of cases of confusion with other activities showed experimental patterns of well-classified action. There were 278/278 lying samples, 228/228 sitting samples, 215/216 standing samples, 286/290 walking samples, and 198/202 jogging samples correctly classified. Although 4/202 jogging samples, 5/291 walking samples, and 2/216 standing samples were misclassified, the error rate in the range of 0.5% - 2% was acceptable.

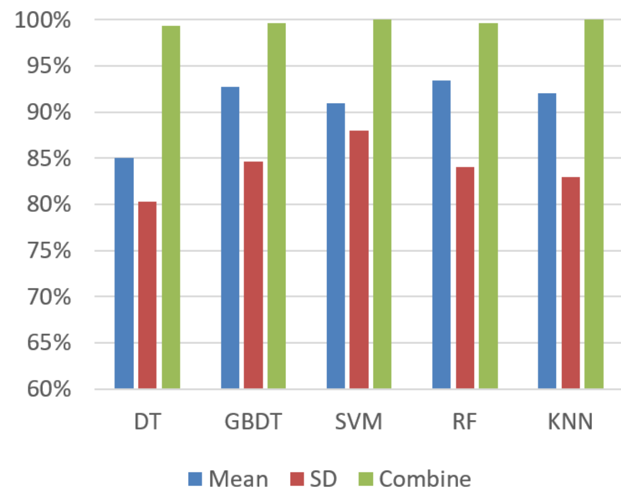

Figure 13. Classification results for algorithms

TABLE VII. Comparison when applying some common algorithms

Indexes	Lying	Sitting	Standing	Walking	Jogging
TP	278	228	215	286	198
FN	0	0	1	6	4
FP	0	0	2	5	4
TN	927	977	990	919	1007
acc (%)	100	100	99.75	99.1	99.34
sen (%)	100	100	99.54	97.95	98.02
spe (%)	100	100	99.80	99.46	99.6

Table VII showed the accuracy, sensitivity and specificity when classifying lying and sitting activities reaching 100%. In general, with the collected dataset, the evaluation indicators were over 97%. It gave information about the decision tree classification model and was very effective because the data of the actions was homogeneous. There was little confusion between the activities being sampled. To evaluate the model's applicability, the research team implemented the classification algorithm (Figure 9) on real devices with the implementation process, as shown in Figure 3.

C. Experimental Results

Classifying activities on the experiment were conducted as follows: each activity was tested in 5 minutes. During the experiment, the activities that the authors observed had a flexible transition between them. The experimental results classifying by the waist-mounted device are synthesized on the confusion matrix (Table VIII). Example: 9/64 jogging was misclassified as standing.

TABLE VIII. Confusion matrix when experimenting on our dataset

Activity	Proposed Activity				
	Lying	Sitting	Standing	Walking	Jogging
Lying	64	0	0	0	0
Sitting	3	61	5	0	0
Standing	0	5	55	2	0
Walking	0	3	7	61	3
Jogging	0	0	0	9	56

There was a certain similarity between standing and sitting positions when the number of samples confused with each other was 5/66 and 5/68, respectively. In addition, some walking activities were misclassified as standing; some jogging activities were mistaken for walking. The performance when classifying each action on the device in real-time was described in Table IX.

TABLE IX. Performance of the model after the experiment

Indexes	Lying	Sitting	Standing	Walking	Jogging
TP	64	61	55	61	56
FN	0	8	7	13	9
FP	3	8	12	11	3
TN	233	236	242	236	241
acc (%)	99	94.89	93.99	92.52	96.12
sen (%)	100	88.41	88.71	82.43	86.15
spe (%)	98.73	96.72	95.28	95.55	98.77

The overall efficiency while classifying on devices was over 92% in terms of accuracy. In particular, the exceptionally high accuracy achieved 99% with lying activity, followed by jogging activity with 96.12%. The standing and sitting activities had slightly less accuracy but were greater than the walking activity. The experimental result illustrated the accuracy of real-time classification was reduced to 8-9% compared to the classification model on the dataset. The lowest accuracy after training and classification with the dataset (Table VII) was 99.1% compared to the lowest accuracy when testing (Table IX), 92.52%. Real-time classification experiments had reduced accuracy because accelerometer data was not homogeneous. The time window of 6 seconds can be mixed up by a few seconds of other activities, making it difficult to classify. For example, the standing activity time segment might exist transition time from walking to standing and vice versa. Still, it has never happened with the computer-trained dataset because it has been grouped by activities. Besides, cumulative noise could occur when switching from static to dynamic at a fast rate.

4. Discussion

The real-time action classification decreased accuracy because the collected accelerometer data was not uniform. In a data segmentation, it was possible to confuse seconds of previous or next activity, making classification difficult. For example, the timing of a standing action might exist as a transition time from the walking to the standing and vice versa. In addition, the limitation of the device was that the sensor must be fixed to the waist, and noise accumulation could occur when switching from static to dynamic at a fast rate. Because the actual data is always noisy, the classification model in this research has given good experimental results and is highly applicable.

In today's modern life, health care [39] needs are increasingly focused on, especially the sick and the elderly. Advanced activity classification applications would be integrated to assist in tracking human activities [40]. Feature extraction played an important role in human activities classification. Many studies selected quite a few features to improve the rate of classification activity. For example, in [11], the authors used 43 features of Catal [17], Min [18], Kwapisz [19], and Van der Maaten [20].

In the document [17], Catal combined many features and gave a result of 91.6%. Although they classified more activities (six activities) than in our study, they tested a more complex algorithm by combining many different classification algorithms. Besides, the use of 43 different features made the cost high, the equipment complex, and difficult to apply on a mass scale. The comparison of our research results with that of Nishkam Ravi [22] and Ling Bao [23] was presented in Table X.

TABLE X. Comparison with Navi and Bao classification models

Activity	L. Bao	N. Ravi	Ours
Lying	95%		99.00%
Sitting	94.8%	96.6%	94.89%
Standing	95.7%	91.8%	93.99%
Walking	89.7%	99.5%	92.52%
Jogging	87.7%	84.8%	96.12%
Total	92.58%	94.54%	95.3%

Our research accuracy when classifying jogging reached 96.12% compared to 87.7% of L. Bao and 84.8% of Ravi. The classification model with lying reached 99% compared to Bao's only 95%. For walking, our classification result (92.52%) was lower than that of N. Ravi (99.5%) but greater than that of L. Bao (89.7%). Overall, this research's classification model using decision trees had the highest average efficiency with 95.3%.

We continued to evaluate the performance of the classification model on the dataset of Kwapisz [19]. Kwapisz's activities data was surveyed, collected from 36 different people with six activities: sitting, standing, walking, jogging, upstairs, and downstairs. This dataset had about 1098207 samples of action data sampled at 20Hz. The name of this dataset is Wireless Sensor Data Mining (WISDM) and acceleration unit used is m/s^2. On this dataset, we reduced the magnitude of the values and converted to unit of g with the conversion rate $g = 9.8m/s^2$. With a sampling frequency of 20Hz, the data collected was very large while the amount of information received was disproportionate. Because the change of activities in about 50ms was very small. Therefore, we reduced the sampling frequency to 1Hz and used ten folds cross-validations [41] on the public dataset. The results obtained were based on an average of 10 runs. We used the public dataset from [19] at a rate of 1 sample per second and window data of 6 seconds to have a fair comparison. Hence, activity window data included 457 of sitting, 371 of standing, 3385 of walking, and 2617 of jogging. Table XI gave information about Mean and SD features at a data window and the number of observation windows corresponding to each activity.

TABLE XI. Features of window pattern on the WISDM [19]

Features	**Axis**	**SIT**	**STAND**	**WALK**	**JOG**
Mean	X(g)	0.328	0.364	1.132	1.128
	Y(g)	0.952	0.951	0.087	0.202
	Z(g)	0.124	-0.204	-0.735	-0.149
SD	X(g)	0.017	0.012	0.654	0.827
	Y(g)	0.006	0.007	0.780	0.635
	Z(g)	0.022	0.014	0.327	0.398
Windows		457	371	3385	2617

Similar to the analyzes in Table IV, the feature values of jogging and walking on this data set were significantly different from those of sitting and standing. The features value gap between two moving and less moving states was significant. In detail, standing and sitting had SD_{max} = $0.022g$ while jogging and walking had SD_{min} = $327g$. The Mean feature on the Z-axis also showed the difference between sitting and standing states. For example, the Z-Mean of sitting is $0.124g$, while standing has a value of $-0.204g$. Thus, the analyzes in Table IV and Table XI had similarities, and the difference was insignificant.

We applied ten cross-validations on the public dataset that the authors [19] used. The average result was 88.75% (from 86.82% to 90.19%). Thus, these results fluctuated slightly, with a deviation of about 1.14%. Table XII showed the best results in ten-fold runs.

A comparison of the results obtained with the results of Kwapisz was presented in Table XIII. The model proposed by the research team had a greater classification efficiency of 98.49% in sitting and 98.9% on standing. With walking and jogging, Kwapisz gave greater results of 91.9% and 98%, respectively. Besides that, he's average sensitivity Index was 91% while ours was a bit greater at 91.4%. In terms of mean specificity, his study showed greater results with 97.1%. He achieved an average good stool score of 95.9% using 43 features and analyzing multiple people's data. However, the investment cost, building equipment for

TABLE XII. Confusion matrix when experimenting on public data

Activity	Proposed Activity			
	Sitting	Standing	Walking	Jogging
Sitting	49	3	1	0
Standing	0	38	2	0
Walking	0	2	317	24
Jogging	0	0	35	212
acc (%)	99.35	98.88	90.59	91.26
sen (%)	92.45	95	92.42	85.83
spe (%)	100	99.14	88.72	94.39

TABLE XIII. Comparison of classification results on results on public data

Activity	**Proposed Activity**			**Ours**		
	acc	sen	spc	acc	sen	spc
Sitting (%)	99.4	93.7	99.7	99.4	92.5	100
Standing (%)	95.9	91	97.1	98.9	95	99.1
Walking (%)	91.9	90.6	92.6	90.6	92.4	88.7
Jogging (%)	98	96.9	98.6	91.3	85.8	94.4
Average (%)	95.9	91	97.1	95	91.4	95.6

data collection and analysis was not small. In this paper, we used two features: Mean and SD, but the average result was 95%. Besides, we aimed to build a compact classification system, low cost, low hardware configuration, and simple embedding algorithm. Our model to be applied to subjects performing hazardous work such as firefighters will be further developed to accept other types of actions such as crawling, upstairs, downstairs, survival status, positioning [42], [43]. Because the activities of these objects often take place with greater intensity in complex conditions such as large fires and dense smoke.

5. Conclusion

This paper built a recognition system with low cost, short computation time, and real-time response. The overall accuracy is 99.3% when training and classifying the activities on the computer. The classification accuracy decreased but was still very good for real-time validation, reaching over 90%. The test results with a sliding window of size 6 seconds gave information about the classification by which the decision tree model had high accuracy. Besides, in addition to the statistical features over time, the study can expand other features in the frequency domain or combine information on both time and frequency domains in the process of feature extraction; while improving these results through experiments on more complex operations to test the accuracy of the proposed model. Experiments are capable of conducting on learning models such as deep learning [44] methods to find the most optimal solution that matches the goal of the classification problem. In addition, we tend to research and analyze complex problems with subjects performing hazardous jobs that need highly interactive support, such as search, rescue [45]–[48] and firefighters [49], [50].

Ethical Approval

Researches and experiments are conducted on volunteers who are students in our department. They all agreed to join the experiment. They are all kept information hidden and personal identity confidential.

Acknowledgment

This research was supported by Hanoi University of Industry (HaUI) [grant number 22 -2021 -RD/H-HCN].

Duc-Nghia Tran was supported by the Domestic Post-doctoral Fellowship Program of Vingroup Innovation Foundation (VINIF), Vingroup Big Data Institute (VinBigdata), code VINIF.2021.STS.21.

References

[1] M. Pham, Y. Mengistu, H. M. Do, and W. Sheng, "Cloud-based smart home environment (coshe) for home healthcare," " in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 483–488, 2016.

[2] K. K. Peetoom, M. A. Lexis, M. Joore, C. D. Dirksen, and L. P. De Witte, "Literature review on monitoring technologies and their outcomes in independently living elderly people," *Disability and Rehabilitation: Assistive Technology*, vol. 10, no. 4, pp. 271–294, 2015.

[3] P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti, "A high reliability wearable device for elderly fall detection," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4544–4553, 2015.

[4] P. Van Thanh, D. T. Tran, D. C. Nguyen, N. Duc Anh, D. Nhu Dinh, S. El-Rabaie, and K. Sandrasegaran, "Development of a Real-Time, Simple and High-Accuracy Fall Detection System for Elderly Using 3-DOF Accelerometers," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3329–3342, 2019.

[5] L. Francés, P. Morer, M. I. Rodriguez, and A. Cazón, "Design and development of a low-cost wearable glove to track forces exerted by workers in car assembly lines," *Sensors*, vol. 19, no. 2, p. 296, 2019.

[6] A. Jain and V. Kanhangad, "Exploring orientation and accelerometer sensor data for personal authentication in smartphones using touch-screen gestures," *Pattern Recognition Letters*, vol. 68, pp. 351–360, 2015.

[7] I. P. E. S. Putra, J. Brusey, E. Gaura, and R. Vesilo, "An event-triggered machine learning approach for accelerometer-based fall detection," *Sensors (Switzerland)*, vol. 18, no. 1, pp. 1–18, 2018.

[8] B. Kamiński, M. Jakubczyk, and P. Szufel, "A framework for sensitivity analysis of decision trees," *Central European Journal of Operations Research*, vol. 26, no. 1, pp. 135–159, 2018.

[9] S. Abudalfa and H. Qusa, "Evaluation of semi-supervised clustering and feature selection for human activity recognition," *International Journal of Computing and Digital Systems*, vol. 8, no. 6, 2019.

[10] S. Begampure and P. Jadhav, "Intelligent video analytics for human action detection: a deep learning approach with transfer learning," *International Journal of Computing and Digital System*, vol. 11, no. 1, pp. 63–71, 2022.

[11] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, and R. Socher, "Deep learning-enabled medical computer vision," *NPJ digital medicine*, vol. 4, no. 1, pp. 1–9, 2021.

[12] H. Luo, C. Xiong, W. Fang, P. E. Love, B. Zhang, and X. Ouyang, "Convolutional neural networks: Computer vision-based workforce activity assessment in construction," *Automation in Construction*, vol. 94, pp. 282–289, 2018.

[13] R. Rodrigues, N. Bhargava, R. Velmurugan, and S. Chaudhuri, "Multi-timescale trajectory prediction for abnormal human activity detection," *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, pp. 2615–2623, 2020.

[14] T. Rahman, A. T. Adams, R. V. Ravichandran, M. Zhang, S. N. Patel, J. A. Kientz, and T. Choudhury, "Dopplesleep: A contactless unobtrusive sleep sensing system using short-range doppler radar," " in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 39–50, 2015.

[15] S. Jamil, A. Basalamah, A. Lbath, and M. Youssef, "Hybrid participatory sensing for analyzing group dynamics in the largest annual religious gathering," " in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 547–558, 2015.

[16] N. C. Minh, T. H. Dao, D. N. Tran, N. Q. Huy, N. T. Thu, and D. T. Tran, "Evaluation of smartphone and smartwatch accelerometer data in activity classification," " in *2021 8th NAFOSTED Conference on Information and Computer Science (NICS)*, pp. 33–38, 2021.

[17] C. Catal, S. Tufekci, E. Pirmit, and G. Kocabag, "On the use of ensemble of classifiers for accelerometer-based activity recognition," *Applied Soft Computing Journal*, vol. 37, pp. 1018–1022, 2015.

[18] Y. Min, Y. Yin, and K. Khin, "Comparing the Performance of Machine Learning Algorithms for Human Activities Recognition using WISDM Dataset," *International Journal of Computer*, vol. 38, pp. 61–72, 2020.

[19] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[20] R. A. Hamad, E. Järpe, and J. Lundström, "Stability analysis of the t-sne algorithm for human activity pattern data," " in *2018 IEEE international conference on systems, man, and cybernetics (SMC)*, pp. 1839–1845, 2018.

[21] T. H. Dao, V. C. Ngo, Q. H. Nguyen, D. N. Tran, and D. T. Tran, "Building human activity recognition system using accelerometers and machine learning methods on low performance microcontrollers," *Research and Development on Information and Communication Technology*, vol. 12/2021, no. 2, pp. 69–76, 2021.

[22] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *Aaai*, vol. 5, no. 2005, pp. 1541–1546, 2005.

[23] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," " in *International conference on pervasive computing*, pp. 1–17, 2004.

[24] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, "The mobiact dataset: Recognition of activities of daily living using smartphones," in *International Conference on Information and Communication Technologies for Ageing Well and e-Health*, vol. 2, pp. 143–151, 2016.

[25] G. Biagetti, P. Crippa, L. Falaschetti, S. Orcioni, and C. Turchetti, "Human activity monitoring system based on wearable sEMG and accelerometer wireless sensor nodes," *BioMedical Engineering Online*, vol. 17, no. S1, pp. 1–18, 2018. [Online]. Available: https://doi.org/10.1186/s12938-018-0567-4

[26] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini, and I. De Munari, "IoT Wearable Sensor and Deep Learning: An Integrated Approach for Personalized Human Activity Recognition in a Smart Home Environment," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8553–8562, 2019.

[27] A. Barna, A. K. M. Masum, M. E. Hossain, E. H. Bahadur, and M. S. Alam, "A study on human activity recognition using gyroscope, accelerometer, temperature and humidity data," " in *2019 international conference on electrical, computer and communication engineering (ecce)*, pp. 1–6, 2019.

[28] P. Cong Phi Khanh, D.-T. Tran, T. Van Duong, N. Hong Thinh, and D.-N. Tran, "The new design of cows' behavior classifier based on acceleration data and proposed feature set," *Mathematical Biosciences and Engineering*, vol. 17, no. 4, pp. 2760–2780, 2020.

[29] D. N. Tran, T. N. Nguyen, P. C. P. Khanh, and D.-T. Trana, "An iot-based design using accelerometers in animal behavior recognition systems," *IEEE Sensors Journal*, 2021.

[30] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 1–33, 2014.

[31] E. E. Abel and A. L. S. Muhammad, "Management of wsn-enabled cloud internet of things: a review," *International Journal of Computing and Digital Systems*, vol. 10, pp. 353–372, 2021.

[32] J. Mesquita, D. Guimarães, C. Pereira, F. Santos, and L. Almeida, "Assessing the esp8266 wifi module for the internet of things," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, pp. 784–791, 2018.

[33] T. H. Dao, T. L. P. Dang, H. H. Tran, and X. T. Tran, "RESEARCH AND DESIGN SMARTLOCKICTU DEVICE," *TNU Journal of Science and Technology*, vol. 13, no. 189, pp. 67–71, 2018.

[34] G. Li, S. Fang, J. Ma, and J. Cheng, "Modeling merging acceleration and deceleration behavior based on gradient-boosting decision tree," *Journal of transportation engineering, Part A: Systems*, vol. 146, no. 7, p. 05020005, 2020.

[35] L. Li, S. Dai, Z. Cao, J. Hong, S. Jiang, and K. Yang, "Using improved gradient-boosted decision tree algorithm based on kalman filter (gbdt-kf) in time series prediction," *The Journal of Supercomputing*, pp. 1–14, 2020.

[36] J. Rani Kaka and K. S. Prasad, "Alzheimer's disease detection using correlation based ensemble feature selection and multi support vector machine," *International Journal Of Computing and Digital System*, vol. 1, no. 1, pp. 9–20, 2021.

[37] D. J. Suroso, A. S. Rudianto, M. Arifin, and S. Hawibowo, "Random forest and interpolation techniques for fingerprint-based indoor positioning system in un-ideal environment," *International Journal of Computing and Digital Systems*, vol. 10, no. 1, pp. 701–713, 2021.

[38] M. Hammad, A. Alqaddoumi, and H. Al-Obaidy, "Predicting software faults based on k-nearest neighbors classification," *International Journal of Computing and Digital Systems*, vol. 8, no. 5, pp. 462–467, 2019.

[39] D. Mahmood, A. Fadhil, and M. M. Rafaa, "Iot based on health caring systems survey," *International Journal of Computing and Digital Systems*, vol. 10, pp. 1–10, 2020.

[40] O. F. Arowolo, E. O. Arogunjo, D. G. Owolabi, and E. D. Markus, "Development of a human posture recognition system for surveillance application," *International Journal of Computing and Digital Systems*, vol. 10, no. 1, pp. 1191–1197, 2021.

[41] F. Y. Ahmed, Y. H. Ali, and S. M. Shamsuddin, "Using k-fold cross validation proposed models for spikeprop learning enhancements," *International Journal of Engineering & Technology*, vol. 7, no. 4.11, pp. 145–151, 2018.

[42] V. T. Pham, Q. B. Le, D. A. Nguyen, N. D. Dang, H. T. Huynh, D. T. Tran *et al.*, "Multi-sensor data fusion in a real-time support system for on-duty firefighters," *Sensors*, vol. 19, no. 21, p. 4746, 2019.

[43] T. D. Tan, L. Ha, N. Long, N. Duc, and N. Thuy, "Integration of inertial navigation system and global positioning system: Performance analysis and measurements," " in *2007 International Conference on Intelligent and Advanced Systems*, pp. 1047–1050, 2007.

[44] I. Idrissi, M. Azizi, and O. Moussaoui, "A Lightweight Optimized Deep Learning-based Host-Intrusion Detection System Deployed on the Edge for IoT," *International Journal of Computing and Digital Systems*, vol. 11, no. 1, pp. 209–216, 2022.

[45] O. M. Alofe, K. Fatema, M. A. Azad, and F. Kurugollu, "Persation: an iot based personal safety prediction model aided solution," *International Journal of Computing and Digital Systems*, vol. 9, no. 6, pp. 1047–1034, 2020.

[46] A. R. Khan and M. A. Chishti, "Data aggregation mechanisms in the internet of things: a study, qualitative and quantitative analysis," *International Journal of Computing and Digital Systems*, vol. 9, no. 2, pp. 289–297, 2020.

[47] D. C. Schedl, I. Kurmi, and O. Bimber, "An autonomous drone for search and rescue in forests using airborne optical sectioning," *Science Robotics*, vol. 6, no. 55, 2021.

[48] Ċ. Mainwaring and D. DeBono, "Criminalizing solidarity: search and rescue in a neo-colonial sea," *Environment and planning C: politics and space*, vol. 39, no. 5, pp. 1030–1048, 2021.

[49] Y. Shi, J. Kang, P. Xia, O. Tyagi, R. K. Mehta, and J. Du, "Spatial knowledge and firefighters' wayfinding performance: A virtual reality search and rescue experiment," *Safety science*, vol. 139, pp. 1–13, 2021.

[50] A. J. Trivisonno, M. R. Laffan, H. K. Giuliani, J. A. Mota, G. R. Gerstner, A. E. Smith-Ryan, and E. D. Ryan, "The influence of age on the recovery from worksite resistance exercise in career firefighters," *Experimental Gerontology*, vol. 152, pp. 1–11, 2021.

Nguyen Thi Thu was born in Ha Noi, Viet nam in 1977. She received her E.Eng. from University of Transport and Communications, Vietnam, in 2000. She received her M.Eng. in Telecommunications Engineering from University of Engineering and Technology, Vietnam in 2005 and Dr.Eng. in Electronic Engineering from Le Quy Don Technical University, Vietnam in 2018. Dr. Nguyen research interests are in the areas of space-time signal processing for communications such as MIMO, spatial modulation and cooperative communications, artificial neural networks.

To-Hieu Dao was a lecturer at University of Information and Communication Technology (ICTU), Thai Nguyen University (TNU). He received his Master degree from Hanoi University of Science and Technology (Vietnam). He has conducted about 6 grassroots-level scientific research projects and guided many student-level research projects. In August 2021, he started researching his doctoral thesis at Vietnam Academy of Science and Technology (VAST). In 2019 and 2021, he received a certificate of merit from the President of Thai Nguyen University. In 2021, he won many awards in scientific research and technology transfer. In 2022, he becomes a lecturer at Phenikaa University.His main research interests include the representation, processing, analysis, and communication of information embedded in signals and datasets. His main research interests include electronics, intelligent systems and IoT.

Bo Quoc Bao received the B.E degrees in electronics and telecommunication from University of Transport and Communications in 2000, the M.S. degree in information processing and communication from Hanoi University of Science and Technonlogy in 2005, and the Ph.D. degree in electrical engineering from Institute of Military Technology, Vietnam in 2019. Since 2000, he has been with Hanoi University of Industry, where he is currently working as the Vice Dean of Faculty of Electronics. His research interests include image processing, machine learning, and computer vision.

Duc-Nghia Tran received the Ph.D. degree from Université Paris Descartes, Paris, France, in 2018. In his thesis, he focuses on signal processing of electron paramagnetic resonance (EPR) spectra for in vivo experiments. He is currently working with the Institute of Information Technology (IOIT), Vietnam Academy of Science and Technology (VAST), Hanoi, Vietnam. His research interests are EPR, IoT, machine learning applications, and data analysis. His publication received the "Best Student Paper Award" at The International Conference on Advanced Technologies for Communications 2017 (ATC'17).

.

Pham Van Thanh was born in 1990. He received the M.Sc. degree in Electronics and Telecommunication at University of Engineering and Technology (UET) in 2012 and he was a PhD student at this university. He also is a full-time lecturer at the Department of Fire protection systems, FFT , University of Fire Prevention and Fighting (UFPF). His research interest covers areas that include Signal and Image Processing, Personal Safety Equipment, Firefighter Supporting Devices, and Fire Protection Systems. He successfully defended his doctoral thesis in 2022.

Duc-Tan Tran is an Associate professor and Vice Dean of Faculty of Electrical and Electronic Engineering, Phenikaa University. His publications received the "Best Paper Award" at the 9th International Conference on Multimedia and Ubiquitous Engineering and International Conference on Green and Human Information Technology (2015). He was the recipient of the award for the excellent young researcher from VNU in 2008, Hanoi, and the third prize in the contest "Vietnamese Talents" in 2008. His main research interests include the representation, processing, analysis, and communication of information embedded in signals and datasets.