Research Article

Academic Platform Journal of Engineering and Smart Systems (APJESS) 11(3), 95-106, 2023

Received: 18-Apr-2022 Accepted: 22-May-2023

homepage: https://dergipark.org.tr/tr/pub/apjess

https://doi.org/10.21541/apjess.1105362

Human Activity Recognition with Smartwatch Data by using Mahalanobis Distance-Based Outlier Detection and Ensemble Learning Methods

*1Ensar Arif SAĞBAŞ, 2Serkan BALLI

[1] Department of Information Systems Engineering, Mugla Sitki Kocman University, Muğla, Türkiye, arifsagbas@mu.edu.tr

[2] Department of Software Engineering, Burdur Mehmet Akif Ersoy University, Burdur, Türkiye, serkanballi@mehmetakif.edu.tr

Abstract

Recognition of human activities is part of smart healthcare applications. In this context, the detection of human activities is an area that has been studied for many years. In these studies, various motion sensors placed in the body are utilized. With the increase in the usage of smart devices, smartphones, and smartwatches have become the constant equipment of these studies thanks to their internal sensors. Sometimes abnormal data are included in data sets due to the way the data were collected and for reasons arising from the sensors. For this reason, it becomes important to detect outlier data. In this study, step counter and heart rate sensors were used in addition to an accelerometer and gyroscope in order to detect human activities. Afterward, the outliers were detected and cleared with a Mahalanobis distance-based approach. With the aim of achieving a better classification performance, machine learning methods were used by strengthening them with ensemble learning methods. The obtained results showed that step counter, heart rate sensors, and ensemble learning methods positively affect the success of the classification. In addition, it was found that the Mahalanobis distance-based outlier detection method increased the classification accuracy significantly.

Keywords: activity recognition; smartwatch; mahalanobis distance; ensemble learning; sensor data

1. INTRODUCTION

Multiple participants, such as doctors and patients, hospitals, and research organizations, are involved in smart healthcare. Disease prevention and monitoring, diagnosis and treatment, hospital management, health-related decision-making, and medical research are all part of this organic whole. Smart health services, for example, are built on the foundation of information technologies such as the Internet of Things, mobile internet, cloud computing, big data, 5G, microelectronics, and artificial intelligence, as well as current biotechnology. These technologies are frequently employed in smart healthcare in all sectors. Patients can utilize wearables to always keep track of their health, receive medical assistance through virtual assistants, and use remote homes to implement remote services; physicians can employ a range of sophisticated clinical decision support systems to aid and improve diagnosis. The adoption of mobile medical platforms can help patients have a better experience. For scientific research institutions, techniques like machine learning can be used instead of manual drug screening, and big data can be used to locate appropriate themes [1].

Recognition of human activities is a useful task in many subjects such as fall detection of elderly people, healthcare applications, and tracking daily routines [2, 3], etc. With the expansion of the usage of wearable sensors, recognition of human activities is possible by using ubiquitous devices, such as smartphones and smartwatches. The increasing popularity of smartwatches facilitates personal health monitoring [4]. These devices have a lot of built-in sensor equipment for instance accelerometer, gyroscope, step counter, etc. With the aid of these sensors, the classification of human activities may be done by using machine learning methods. However, sometimes these sensor signals are exposed to effects such as noise. This situation causes a bad fit between the data and includes it in the general pattern calculation [5]. Clearing outlier data is an important preprocessing step to create more consistent models. Ensemble methods are also used in activity recognition. Ensembles classifiers train more than one base learner instead of the single base learner. Thus, these methods can contribute to increasing the accuracy rate.

In this study, step counter and heart rate sensors were used in addition to the accelerometer and gyroscope sensors to create a new dataset. Dataset consists of walking, jogging, writing on paper, writing on the blackboard, typing, stationary, vacuuming, and brushing teeth activities. Outlier data from the created data set was cleared with a structure

Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/

built based on Mahalanobis distance. This approach based on the detection and cleaning of outliers and the combination of multiple sensors used is the main novelty of this study. Then performances of sensor data combinations were evaluated by using machine learning methods and ensemble learning approaches. Machine learning methods and ensemble approaches which are frequently encountered in the literature and have high success rates were chosen [6]. The most accurate result was obtained from the Random Subspace ensemble of the kNN method with the accelerometer, gyroscope, heart rate, and step counter sensors combination. Step counter and heart rate sensors increased the success rate.

The paper is organized as follows: previous studies will be mentioned in section two. After that machine learning methods and ensemble learning approaches, the creation of the dataset, and Mahalanobis distance-based outlier detection will be described, used sensors clarified in Section three. Afterward, the experimental results of the study will be handled in Section four. In the continuation, the impact of Mahalanobis distance-based outlier detection on classification will be discussed in Section five. Eventually, section six will conclude the paper.

2. RELATED WORK

While examining the related works, various studies were found about activity recognition by using machine learning methods and wearable devices. Asarakaya and Ünsal [7] aimed to define human activities using machine learning methods on data obtained from smart sensors. Sağbaş and Balli [8] detected the transportation modes (traveling by bus or car, cycling, running, and walking) of the users by using smartphone sensors (accelerometer, gyroscope, and GPS). Six different machine learning techniques (Bayesian Network, Naive Bayes, kNN, Random Forest, J48, and Logistic Regression) were tested and their performances were compared. Erin et al. [9] performed the detection of human activities based on the internet of things by using the accelerometer sensor of the device with the android software developed for the mobile device. Voicu et al. [10] classified six human activities in eight different scenarios by using smartphone sensors. Balli et al. [2] proposed a mobile solution for the detection of falls using together smartwatches and smartphone sensors. Peker et al. [11] predicted human activities with the data obtained from the smartwatch. They first applied the ReliefF attribute selection and then classified them with the Kernel-Based Extreme Learning Machine method. Ahmed et al. [12] proposed a hybrid feature selection method for human activity detection. Yahaya et al. [13] proposed an approach to identify sources of anomalies in human activities. Li et al. [14] proposed a new method of feature extraction based on linear predictive analysis (LPA) to reduce the computational complexity in activity classification using acceleration signals. Gani et al. [15] offered a computationally efficient, smartphone-based human activity recognition system based on chaos theory and dynamic systems. Elsts et al. [16] proposed an energy-efficient activity recognition framework with two key components by using a wearable accelerometer.

In addition, it is possible to come across various studies using deep learning methods. Challa et al. [17] used a hybrid of CNN and BiLSTM models to design a robust classification model for human activity recognition using wearable sensor data. Metin and Karasulu [18] compared the performance of deep learning techniques to classify daily human activities. Munez-Organero [19] proposed an outlier detection algorithm based on Deep recurrent neural networks for detecting human activities. Zhou et al. [20] proposed a Convolutional Neural Network-based structure to detect nine indoor human actions from smartphone sensor data. Wan et al. [21] designed a smartphone accelerometer and deep learning-based architecture for human activity detection. Zhou et al. [22] designed a semi-supervised deep learning framework that efficiently uses weakly labeled sensor data in activity detection. Altuve et al. [23] classified six different human activities using bidirectional LSTM. Mukherjee et al. [24] determined human body movements by using data obtained from smart device sensors and a collection of three classification models, namely CNN-Net, Encoded-Net, and CNN-LSTM, which are called EnsemConvNet.

Ensemble methods were also investigated in various studies. Catal et al. [25] investigated the power of the ensemble of classifiers approach for accelerometer-based activity recognition and built a novel activity estimation model grounded on machine learning classification methods. Elamvazuthi et al. [26] tested five different ensemble learning methods for classifying six daily activities. They gained inertial sensor data from smartphones. Balli et al. [3] proposed a hybrid structure using principal component analysis and Random Forest methods for activity recognition with smartwatches. Herrera-Alcantara et al. [27] observed the activities of students with smartwatches. They obtained the most satisfactory result from the Random Forest method. Irvine et al. [28] proposed a new neural network ensemble method that is aiming to improve the human activity recognition dataset. The ensemble-based approach to detecting human activity was discussed in detail in the study conducted by Brajesh and Ray [29]. Sekiguchi et al. [30] increased the classification success of activity detection with an ensemble model that includes a CNN model and a gradient-boosting model. Subasi et al. [31] used the Adaboost ensemble to classify human activities. Dwivedi et al. [32] introduced a new skeleton-based feature for human activity recognition and used it to train the Random Forest classifier.

This study differs from the other study in the literature with the used sensor combinations. In addition, the effect of outlier data cleaning on classification success was investigated. High classification accuracies were achieved with classification ensemble learning-based approaches.

3. MATERIALS AND METHODS

In this section, the machine learning methods and ensemble learning approaches, the smart device sensors used in the study, the creation of the data set, and Mahalanobis distance-based outlier detection will be explained.

3.1. Machine Learning Methods

Machine learning methods were successfully used in human activity recognition in previous research. Machine learning methods construct an algorithm and make predictions from the dataset. In this section, the machine learning methods used in this study are summarized.

3.1.1. Naïve Bayes (NB)

The Naive Bayes Classifier is a simple probability-based algorithm with a strong assumption of attribute independence. The Naive Bayes Classifier performs learning through test data and incorporates the best proportion of the instance into the class [33, 34]. The Bayes theorem given in Equation 1 is used to estimate the class based on the test data.

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)} \quad (1)$$

Here, P(A|B) is the probability that event A will occur when event B is known. P(B|A) is the probability of event B occurring when event A is known. P(A) is the probability of occurrence of event A. P(B) is the probability of occurrence of event B.

3.1.2. k Nearest Neighbor (kNN)

kNN is based on the similarity between sample data and dataset. The number of nearest neighbors to be considered for the classification in the kNN algorithm is expressed as a positive integer such as k. In determining the closest neighbors, the closeness between the samples in the training set and the selected sample is determined. The closeness between the samples is sorted in ascending. It presents the order from the nearest neighbor to the farthest neighbor of the selected sample [35, 36]. The euclidian distance formula in Equation 2 was used in this study to calculate distance.

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \quad (2)$$

Here p and q represent points on the hyperplane of the data.

3.1.3. J48

J48 is the Java implementation of the C4.5 algorithm. The C4.5 algorithm consists of the classification trees of attributes between numerical and categorical values. It is important which attribute to start branching when creating classification trees. Using a training data set to reveal all possible tree structures and choose the most suitable one among these tree structures causes a lot of repetitions. Therefore, classification tree algorithms are going to create trees according to these values by calculating various values at the beginning of the process. For this purpose, the concept of entropy can be applied. The branching of the tree starts by considering the value of entropy [25, 37]

3.1.4. Random Forest (RF)

Breiman [38] presented the idea of using a huge number of tree structures instead of an individual tree, that is, the use of a forest for classification purposes. In the Random Forest method, samples are selected from the dataset by the Bootstrap method. Then classification trees are generated based on these samples. Using these classification trees, the class of observation is estimated for each tree and the most repeated class value is selected among the classifications [39]. Due to the tree data structure, J48 and Random Forest methods are harder to implement and take more time than other methods for activity recognition applications.

3.2. Ensemble Methods

Four ensemble methods were compared in this study. These methods are AdaBoostM1, Bagging, Random Subspaces, and Voting. In this section, these methods are briefly described.

3.2.1. AdaBoostM1

AdaBoost (Adaptive Boosting) is a popular ensemble algorithm. It was presented in 1996. This algorithm focuses on patterns that are harder to classify. In each iteration of the algorithm, the weights of misclassified samples are increased, and the weights of correctly classified samples are reduced [40].

3.2.2. Bagging

The term Bagging (Bootstrap AGGregatING) is introduced by Breiman [41]. Bagging is a simple and appealing idea. This idea builds a new classifier from the training set. In this case, some samples in the dataset are not included in the newly created training set, while others are found more than once. Each basic classifier in the ensemble structure is trained with these training sets that contain different examples. The output of the classifier is combined with plurality voting to conclude [42].

3.2.3. Random Subspaces

The Random Subspaces algorithm is an ensemble method that randomly selects several components of the given feature vector in constructing each classifier. The purpose of this algorithm is to avoid overfitting while boosting the predictive performance [6].

3.2.4. Voting

Voting is a simple method that combines basic learning algorithms. There are various methods of combining the outputs of basic classification algorithms. These hybrid methods include majority voting and weighted majority voting. In simple majority voting, the result decided to receive the most votes among the k basic classification algorithms is determined as the output of the ensemble classifier [6, 43].

3.3. Dataset and Feature Extraction

Step counter, heart rate monitor, three axes accelerometer, and gyroscope sensor data acquired from the smartwatch (Moto 360) in Figure 1 were used in this study. All data were collected while the smartwatch is attached to the wrist of the user. This device has 512 MB RAM, a quad-core 1.2 GHz processor, and a built-in gyroscope, step counter, accelerometer, optical heart rate monitor, and ambient light sensors. It was set to collect 50 samples per second.

Figure 1. Smartwatch that used in this study

All sensor data were labeled during the data collection phase and split into two seconds size of windows. 500 patterns which include 100 sample sensor data were obtained for each activity: jogging, walking, brushing teeth, writing on paper, typing, writing on a blackboard, stationary, and vacuuming. Jogging means running at a slower velocity. Especially, heart rate value varies in jogging activity and running activity. Stationary includes sitting and standing activities. Using the keyboard means working at a computer and typing with the keyboard.

In the dataset, each activity has the same number of samples. The average value of heart rate, accelerometer, and gyroscope sensors, the standard deviation of accelerometer and gyroscope sensors, and the number of steps (total 14 features) form a pattern. The list of all features and their ranges is given in Table 1.

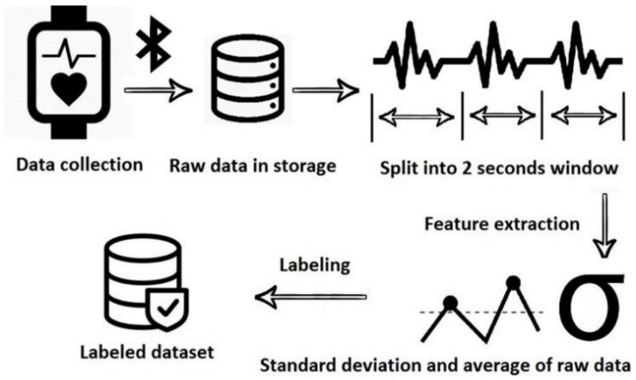

Figure. 2. Dataset lifecycle

Table 1. The list of all features and their ranges

Feature name	Range
Average value of accelerometer X	0.0004-10.7696
Average value of accelerometer Y	0.0043-11.6919
Average value of accelerometer Z	0.0028-0.9369
Standard deviation value of accelerometer X	0.0055-12.2600
Standard deviation value of accelerometer Y	0.0098-16.4635
Standard deviation value of accelerometer Z	0.0053-6.4985
Average value of gyroscope X	0.0000-1.1898
Average value of gyroscope Y	0.0000-1.3245
Average value of gyroscope Z	0.0000-0.8611
Standard deviation value of gyroscope X	0.0034-4.9206
Standard deviation value of gyroscope Y	0.0012-2.3982
Standard deviation value of gyroscope Z	0.0008-2.6774
Number of steps	0-20
Average value of heart rate	0-180.83

Before extracting the features, raw data were not filtered, and all variables were calculated per two second window. Through two-second window, the developed activity recognition system continues to accurately identify activities if the beginning is missed. The dataset lifecycle is given in Figure 2.

3.4. Sensors

To detect human activities based on wrist motion; an accelerometer, gyroscope, step counter, and heart rate sensors were used. This section briefly describes these sensors.

3.4.1. Accelerometer

Acceleration force is applied to a device on the x, y, and z axes (Figure 3), including the force of gravity. The accelerometer measures the acceleration force in m/s2 [44].

Figure. 3. Smartwatch accelerometer axes

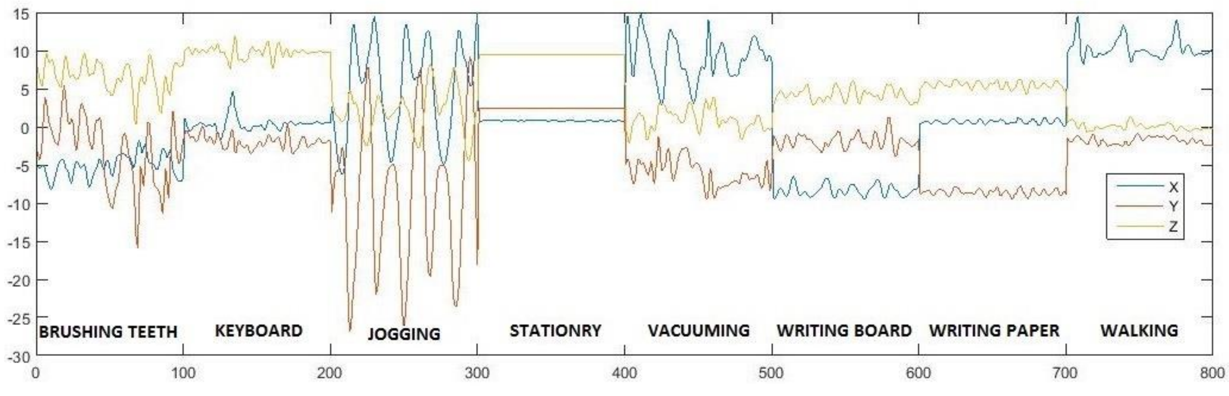

Figure 4. Amplitude changes of accelerometer axes

Figure 4 shows the amplitude change of accelerometer axes for eight different daily activities (brushing teeth, using the keyboard, jogging, stationary, vacuuming, writing on the blackboard, writing on paper, and walking).

In this study, the standard deviation and the average value of accelerometer data were selected as acceleration features. 3-D representations of standard deviations of accelerometer data for each activity are shown in Figure 5.

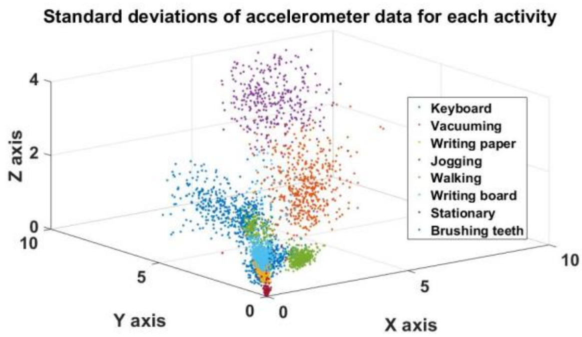

Figure 5. Standard deviations of accelerometer data for each activity

3.4.2. Gyroscope

The gyroscope detects the roll, pitch, and yaw motions of the devices along the x, y, and z-axes, respectively and it calculates the device's rotation rate. The axes' directions are shown in Figure 3. The raw data stream from a gyroscope sensor is the rate of the rotation around each of the three physical axes in rad/s (radian per second) [44]. In this study, the standard deviation and the average value of gyroscope data were used. Figure 6 shows the y-axis amplitude change of the gyroscope sensor for eight different daily activities.

3.4.3. Step Counter

The step counter sensor returns the number of users' steps since the last reboot while activated. The value is returned as a float (with the fractional part set to zero) and is reset to zero only on a system restart. The timestamp of the event is set to the time when the last step for that event was taken [3]. The average number of steps for each activity is given in Figure 7.

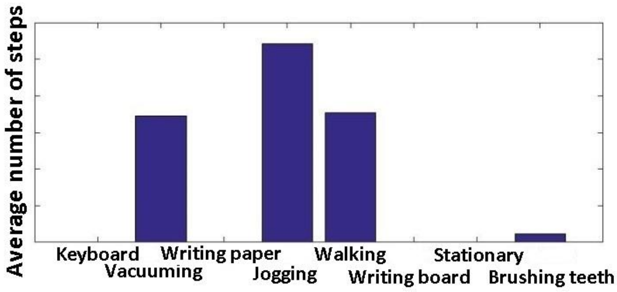

Figure 7. Average number of steps for each activity

3.4.4. Heart Rate Monitor

The reported value is the heart rate in beats per minute. The reported accuracy represents the status of the monitor during the reading [2]. The heart rate sensor rarely returns 0 when the signal is distorted. Average heart rates for each activity are given in Figure 8.

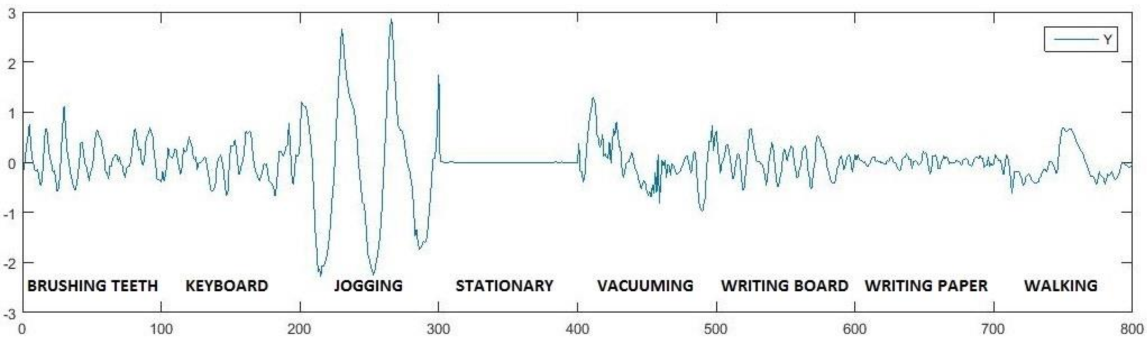

Figure 6. Amplitude changes of gyroscope y-axis

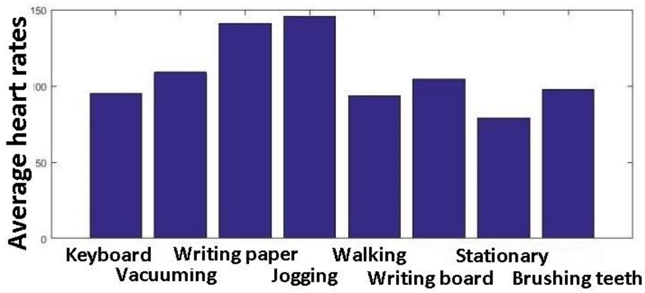

Figure 8. Average heart rates for each activity

3.5. Mahalanobis Distance-Based Outlier Detection

In an ellipse, some points are closer to the center than others (Figure 9), but it cannot be concluded that the more distant points belong less than the points closer to the sample, since this is part of the basic model of the normal distribution. Therefore, instead of the classical distance, it is recommended to use a distance that considers the shape of the observations under investigation, and such a distance is the Mahalanobis distance denoted by d [45]:

$$d = \sqrt{(x - m)^T C^{-1} (x - m)} \tag{3}$$

where x is a vector of variables $x = (x_1, x_2, \ldots, x_k)$, $m = (m_1, m_2, \ldots, m_k)$ is a k-dimensional vector and C is a $k \times k$ symmetric matrix. It measures the distance from a point x to the center of m in metric C, meaning that the distance depends on the shape. Naturally, the values of m and C are practically unknown and therefore need to be estimated.

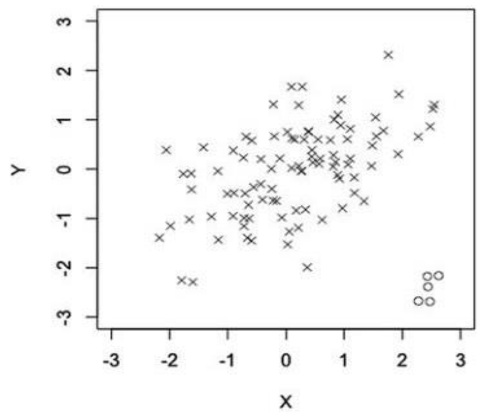

Figure 9. Scatter plot of two variables X, Y sampled from the normal distribution and 5 outliers (circles) [46]

In this study, the data were considered class by class in clearing out the outliers. Mahalanobis distances were calculated to detect outliers. Then, the probability values (1-ChiSquare) were calculated with the SPSS program. The calculated probability, those below the 0.001 threshold, were marked as outliers and deleted from the data set. After clearing the outliers, the number of patterns and cleared data ratios belonging to the classes are presented in Table 2.

It is seen that approximately 4% of the data for each class was selected as outlier data and cleared from the data set. It is striking that the rate of outlier data pertaining only to writing on the board is higher than in the other classes. The size of the dataset decreased by 4.4% after the cleaning process.

The outliers in the sensor data are caused by the participant's out-of-class movements and noisy data. Considering that the step counter sensor is also a multiple motion detector, it is likely to be affected by noisy data. When Table 1 in Section 3.3 is examined, it is seen that the lowest value of the heart rate is 0. This indicates that erroneous readings of the heart rate sensor are also present. In summary, it can be said that there is a certain amount of outlier data in all sensor data.

Table 2. Number of patterns belonging to classes after outliers are cleared

Class name	Number of patterns	Cleared data ratio
Brushing teeth	481	3.8%
Writing on the paper	479	4.2%
Writing on the board	452	9.6%
Walking	484	3.2%
Vacuuming	477	4.6%
Stationary	479	4.2%
Keyboard	486	2.8%
Jogging	485	3.0%

4. EXPERIMENTAL RESULTS

To compare the performance metrics, six different types of sensor combinations (Table 3) were tested with machine learning and ensemble methods. A Diagram of the system architecture is shown in Figure 10. k-fold cross-validation was employed to improve the performance of the used learning method. In cross-validation, the dataset is split into k groups, and the method is employed in each group. In each trial, one of the k-groups is selected as the test set and the other k-1 groups are used to create a training set. Then the mean error through all k trials is calculated [47]. With this approach, all samples in the dataset are used both in the testing phase and in the training phase.

Naive Bayes, kNN, J48, and Random Forest, which are frequently encountered in the literature, were used as base classification methods. In ensemble construction, four ensemble methods (AdaBoost M1, Bagging, Random Subspaces, and Vote) were analyzed. In the voting decision step, NB, kNN, J48, and Random Forest algorithms were chosen, and the average of probabilities combination rule was selected. The experiments were applied with WEKA [48] toolkit version 3.8. The results of the tests were compared with classification accuracy. Classification accuracy rates of methods for datasets obtained without cleaning outliers are given in Table 4 and Figure 11. According to tests carried out, obtaining the best results is displayed in bold.

Table 3. Model structure and input variables for the six models

Model No	Model Name	Contents
1	Accelerometer only	Standard deviation and average value of x, y and z axes accelerometer sensor data (6 features)
2	Accelerometer and Gyroscope	Standard deviation and average value of x, y and z axes accelerometer and gyroscope sensor data (12 features)
3	Accelerometer, Gyroscope and Step Counter	Standard deviation and average value of x, y and z axes accelerometer and gyroscope sensor data and number of steps (13 features)
4	Accelerometer, Gyroscope, Step Counter and Heart Rate	Standard deviation and average value of x, y and z axes accelerometer and gyroscope sensor data, number of steps and average heart rate (14 features)
5	Accelerometer and Step Counter	Standard deviation and average value of x, y and z axes accelerometer sensor data and number of steps (7 features)
6	Accelerometer, Step Counter and Heart Rate	Standard deviation and average value of x, y and z axes accelerometer sensor data, number of steps and average heart rate (8 features)

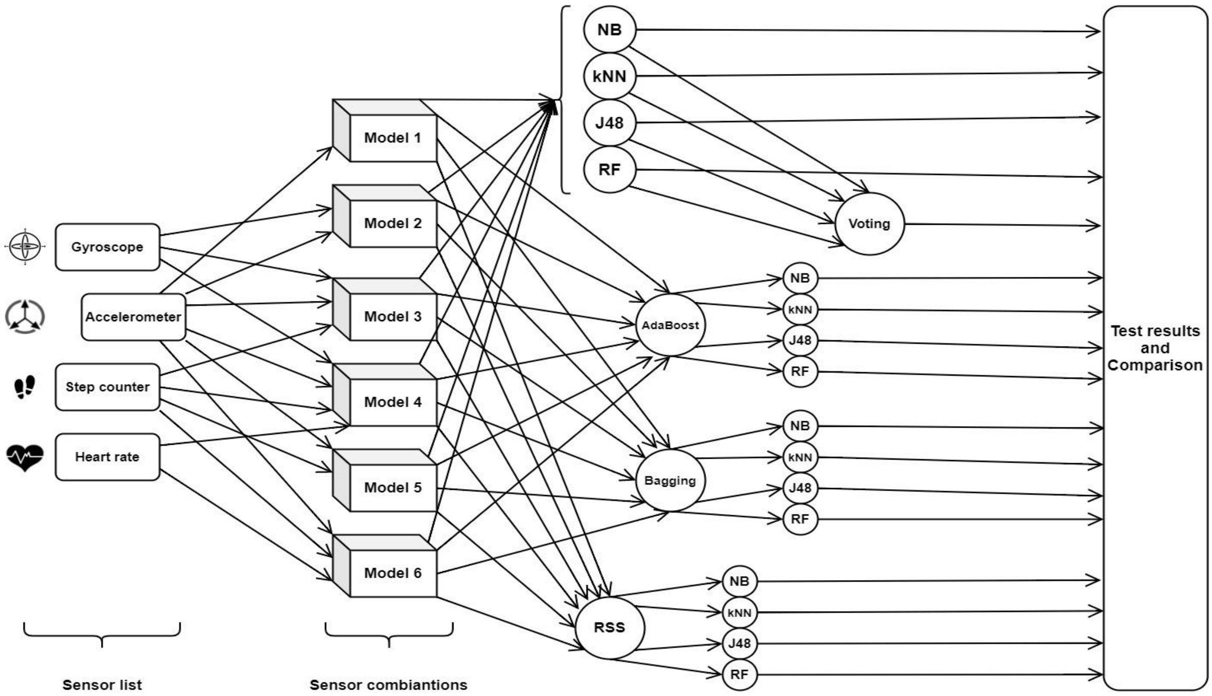

Figure 10. Diagram of the system architecture

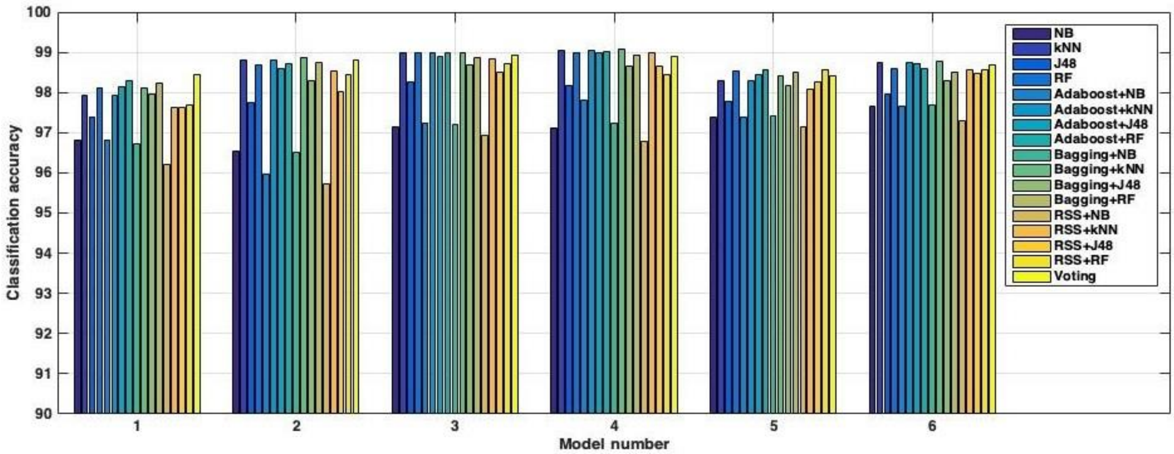

Figure 11. Bar chart of results from the dataset without cleaning outliers

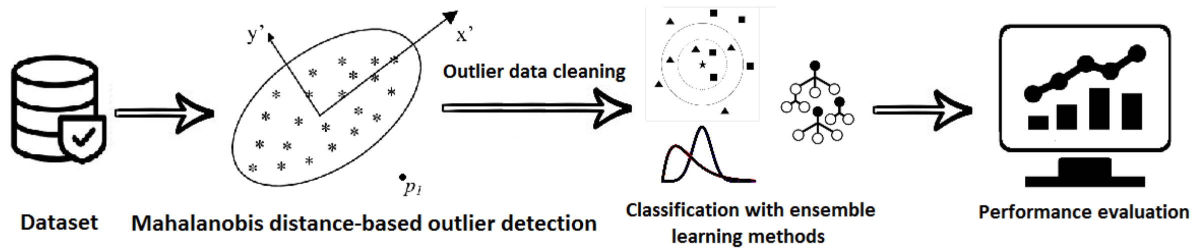

Figure 12. Flow chart of outlier data cleaning and performance evaluation stage

Six different types of sensor combinations were compared in this study, and it is seen that the best classifications based on methods are mostly obtained by using the Model 4 sensor combination (Accelerometer, gyroscope, step counter, and heart rate). It is noteworthy that the accelerometer and step counter have an important place in classification performance. The most successful method was found to be the Bagging ensemble of kNN with an accuracy rate of 99.075%. Thus, it can be said that the heart rate sensor also has a positive effect on classification success. The lowest classification accuracy (95.725%) was obtained from Model 2 (Accelerometer and Gyroscope) with RSS + NB. When comparing Model 2 with Model 1, the accuracy rates of all methods with NB combinations decreased. However, an increase was observed in other methods.

The tests performed with the ensemble learning methods were re-performed after clearing the outliers and their performance was compared. The flow chart of the outlier data cleaning and performance evaluation stage is shown in Figure 12. The accuracy rates for the dataset obtained with outlier data cleared are shown in Table 5 and Figure 13.

Table 4. Classification accuracy rates for dataset obtained without cleaning outliers

Method/Model	1	2	3	4	5	6
NB	96.825	96.550	97.150	97.125	97.375	97.650
kNN	97.925	98.800	98.975	99.050	98.300	98.750
J48	97.375	97.750	98.250	98.175	97.775	97.950
RF	98.125	98.700	99.000	99.000	98.525	98.600
AdaBoost+NB	96.825	95.975	97.250	97.800	97.375	97.650
AdaBoost+kNN	97.925	98.800	98.975	99.050	98.300	98.750
AdaBoost+J48	98.150	98.600	98.900	99.000	98.450	98.725
AdaBoost+RF	98.300	98.725	98.975	99.025	98.550	98.600
Bagging+NB	96.725	96.500	97.200	97.225	97.425	97.675
Bagging+kNN	98.100	98.875	98.975	99.075	98.425	98.775
Bagging+J48	97.950	98.300	98.700	98.650	98.175	98.300
Bagging+RF	98.225	98.750	98.875	98.925	98.500	98.500
RSS+NB	96.200	95.725	96.925	96.775	97.150	97.300
RSS+kNN	97.625	98.525	98.825	98.975	98.075	98.575
RSS+J48	97.625	98.025	98.500	98.650	98.275	98.475
RSS+RF	97.700	98.450	98.725	98.450	98.550	98.550
Voting	98.450	98.800	98.925	98.900	98.425	98.700

According to Table 5, the best performance (99.686%) was obtained by the Model 4 sensor combined with the Random Subspace ensemble of the kNN method. It was observed that only the accelerometer sensor provides a high success rate (minimum 97.489%). Step counter and heart rate sensors and ensemble methods contribute to increasing the accuracy. Model 6 (Accelerometer + Step counter + Heart rate) provided the best result in 9 of the 17 classification methods. When Model 1 and Model 2 were compared, there was no increase in the classification success of only 4 methods. Three classification methods provided the most successful results with Model 3 (Accelerometer + Gyroscope + Step counter). There was not any method that achieved the most successful result with Model 5. These comparisons highlight the importance of the gyroscope sensor. However, the most successful classifications include step counter and heart rate sensors. The positive effects of the new sensors tested are supported by these experiments. In addition, obtaining the highest classification accuracy rate with the Random Subspace ensemble shows that not all the extracted features are efficient.

Table 5. Classification accuracy rates for dataset obtained with outlier data cleaned

Method/Model	1	2	3	4	5	6
NB	98.012	98.117	98.484	98.640	98.718	98.771
kNN	99.111	99.425	99.503	99.582	99.215	99.608
J48	98.666	98.535	98.928	98.849	98.928	98.954
RF	99.346	99.503	98.247	99.608	99.608	99.634
AdaBoost+NB	98.666	99.058	99.163	99.163	99.032	98.849
AdaBoost+kNN	99.111	99.425	99.503	99.582	99.215	99.608
AdaBoost+J48	99.372	99.372	99.425	99.503	99.398	99.320
AdaBoost+RF	99.398	99.582	99.582	99.660	99.608	99.608
Bagging+NB	97.934	98.064	98.509	98.666	98.666	98.771
Bagging+kNN	99.163	99.320	99.477	99.529	99.268	99.555
Bagging+J48	98.980	99.006	99.032	99.006	98.928	99.111
Bagging+RF	99.425	99.555	99.608	99.555	99.477	99.529
RSS+NB	97.489	96.940	98.718	98.666	98.404	98.666
RSS+kNN	98.980	99.582	99.582	99.686	99.241	99.425
RSS+J48	99.032	99.268	99.425	99.555	99.425	99.425
RSS+RF	99.215	99.503	99.634	99.660	99.477	99.503
Voting	99.451	99.451	99.451	99.425	99.294	99.503

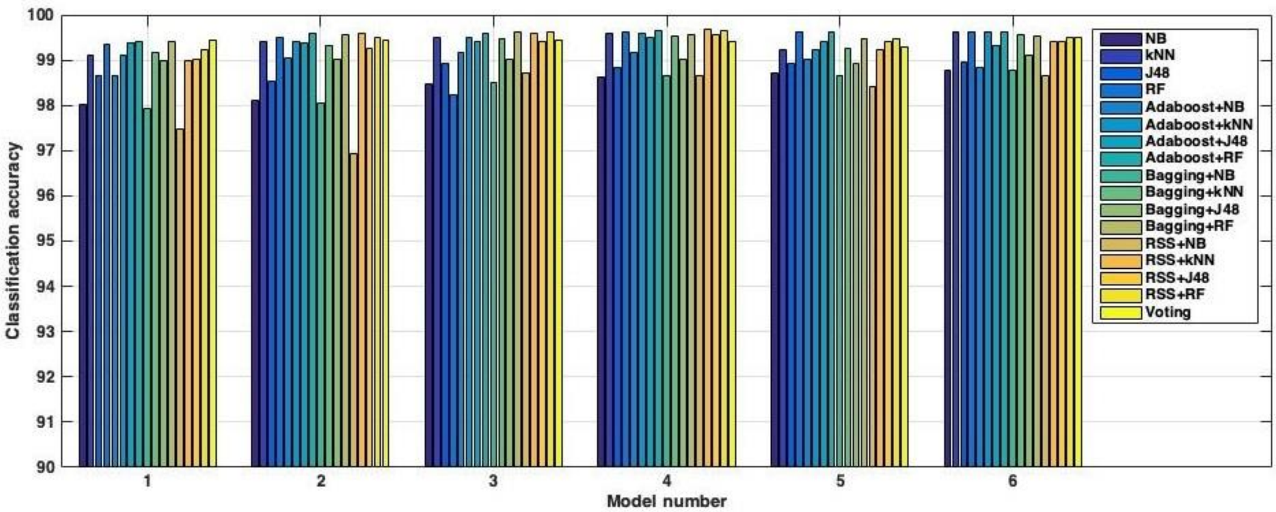

Figure 13. Bar chart of results from the dataset with outlier data cleaned

5. DISCUSSION

In this section, the effect of Mahalanobis distance-based outlier detection and ensemble learning approaches on classification performance will be discussed. When Table 4 and Table 5 in Section 4 are examined, the effect of sensor data on classification success can be seen clearly. The best results were obtained with Model 3, Model 4, and Model 6. All these Models include a step counter. The Model 4 and Model 5 include a heart rate sensor in addition to the step counter. The improvement rates are presented in Table 6 and Figure 14 to better understand the extent to which the outlier data cleaning affects the classification success.

When the differences between classification accuracies are examined, it is seen that only 1 of the 102 experiments performed decreased. An increase was observed in all other experiments. The highest increase was provided by AdaBoost + NB method and Model 2. Mahalanobis distance-based outlier detection and cleaning increased the classification accuracy by an average of 1 point. An N-way analysis of variance test was used to determine statistical significance. The attained p-value given across the multiple classification methods was p 0.01 at a significance threshold of 0.05, indicating substantial differences in the accuracy rate achieved by the different classification methods. The confusion matrix for Model 4 sensor combination with Random Subspace ensemble of kNN is given in Table 7.

According to Table 7, recognition accuracies for walking, writing on paper, and jogging were 100%. Writing on the board activities was confused with brushing teeth and brushing teeth activities were confused with writing on board and vacuuming. One of the vacuuming and stationary activities was misclassified as walking. Because the user was performing these activities in a standing position like walking.

It is not possible to directly compare this study, which was carried out by creating a new data set, with other studies in the literature. In addition to motion sensors, heart rate sensors were used in the study. Considering the total number of samples in the dataset, it was thought that it would be appropriate to use classical machine learning methods instead of deep learning methods. The ensemble learning approach applied after outlier data cleaning provided high-accuracy classification success. But the main drawback of the proposed method is that ensemble learning methods (they have a sequential or parallel operation) need a long computation time and relatively much memory.

Table 6. Improvement rates after outlier data cleaning

Method/Model	1	2	3	4	5	6
NB	1.187	1.567	1.334	1.515	1.343	1.121
kNN	1.186	0.625	0.528	0.532	0.915	0.858
J48	1.291	0.785	0.678	0.674	1.153	1.004
RF	1.221	0.803	-0.753	0.608	1.083	1.034
AdaBoost+NB	1.841	3.083	1.913	1.363	1.657	1.199
AdaBoost+kNN	1.186	0.625	0.528	0.532	0.915	0.858
AdaBoost+J48	1.222	0.772	0.525	0.503	0.948	0.595
AdaBoost+RF	1.098	0.857	0.607	0.635	1.058	1.008
Bagging+NB	1.209	1.564	1.309	1.441	1.241	1.096
Bagging+kNN	1.063	0.445	0.502	0.454	0.843	0.78
Bagging+J48	1.03	0.706	0.332	0.356	0.753	0.811
Bagging+RF	1.2	0.805	0.733	0.63	0.977	1.029
RSS+NB	1.289	1.215	1.793	1.891	1.254	1.366
RSS+kNN	1.355	1.057	0.757	0.711	1.166	0.85
RSS+J48	1.407	1.243	0.925	0.905	1.15	0.95
RSS+RF	1.515	1.053	0.909	1.21	0.927	0.953
Voting	1.001	0.651	0.526	0.525	0.869	0.803

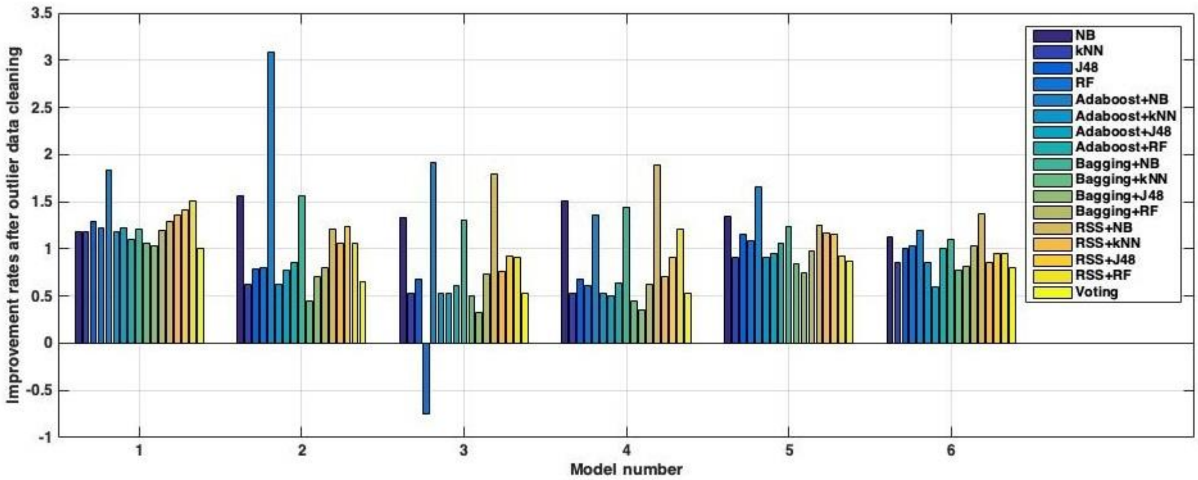

Figure 14. Bar chart of the improvement rates after outlier data cleaning

Table 7. Confusion matrix of Model 4 sensor combination with Random Subspace ensemble of kNN

Classified as	a	b	c	d	e	f	g	h	%
a=Brushing teeth	477	0	2	0	2	0	0	0	99.17
b=Writing on the paper	0	479	0	0	0	0	0	0	100
c=Writing on the board	3	0	448	0	1	0	0	0	99.12
d=Walking	0	0	0	484	0	0	0	0	100
e=Vacuuming	1	0	0	1	475	0	0	0	99.58
f=Stationary	0	0	0	1	0	478	0	0	99.79
g=Keyboard	0	1	0	0	0	0	485	0	99.79
h=Jogging	0	0	0	0	0	0	0	485	100

6. CONCLUSION

This study presents an effective outlier data clearing and ensemble learning-based approach to the recognition of human activities using smartwatch sensors. With the Mahalanobis distance-based outlier detection, approximately 4% of the total data was detected as an outlier and cleared from the data set. Various ensemble approaches were tested in past studies. In this study, four different ensemble approaches were tested and the Random Subspace ensemble of the kNN method achieved the most successful result for activity recognition by using the smartwatch. In addition, step counter and heart rate sensor data performances are investigated in this paper. These sensors also increase accuracy. According to the dataset used in the study, each ensemble method increases the success rate of different sensor combination models. The best result is obtained from the Model 4 sensor combined with the Random Subspace ensemble of the kNN method between all test options. The highest accuracy rate of 99.686% among all test options was obtained from this method. This demonstrates that all the sensors used in the study contribute to the classification and the RSS approach increases the classification success of the kNN method. In future works, this activity recognition study can be improved in various ways such as by increasing the number of classes (e.g.: handshake, smoking, cooking, and drinking, etc.) and expanding the dataset by collecting sensor data from different users since especially heart rate value may vary from person to person.

Author contributions: All authors contributed equally to the creation of the idea and the design. The mobile application used in the study was developed by the first author. The second author contributed to checking the article in terms of spelling and content.
Conflict of Interest: No conflict of interest was declared by the authors.
Financial Disclosure: This study is supported by Mugla Sitki Kocman University Scientific Research Projects under the grant number 016-061.

REFERENCES

[1] S. Tian, W. Yang, J.M. Le Grange, P. Wang, W. Huang, and Z Ye, “Smart healthcare: making medical care more intelligent”, Global Health Journal, vol. 3, no. 3, pp. 62-65, 2019. https://doi.org/10.1016/j.glohj.2019.07.001

[2] S. Ballı, E.A. Sağbaş, and M. Peker, “A mobile solution based on soft computing for fall detection”. In Mobile Solutions and Their Usefulness in Everyday Life, Ed. Sara Paiva, 2019, ch. 14, pp. 275-294.

[3] S. Balli, E.A. Sağbaş, and M. Peker, “Human activity recognition from smart watch sensor data using a hybrid of principal component analysis and random forest algorithm”, Measurement and Control, vol. 52, no. 1-2, pp. 37-45, 2019. https://doi.org/10.1177/0020294018813692

[4] B. Mortazavi, S. Nyamathi, S.I. Lee, T. Wilkerson, H. Ghasemzadeh, and M. Sarrafzadeh, “Near-realistic mobile exergames with wireless wearable sensors”, IEEE Journal of Biomedical and Health Informatics, vol. 18, no. 2, pp. 449-456, 2013. https://doi.org/10.1109/JBHI.2013.2293674

[5] M.C. Limas, J.B.O. Meré, F.J.M. de Pisón Ascacibar, and E.P.V. González, “Outlier detection and data cleaning in multivariate non-normal samples: the PAELLA algorithm”, Data Mining and Knowledge Discovery, vol. 9, no. 2, pp. 171-187, 2004. https://doi.org/10.1023/B:DAMI.0000031630.50685.7c

[6] A. Onan, S. Korukoğlu, and H. Bulut, “Ensemble of keyword extraction methods and classifiers in text classification”, Expert Systems with Applications, vol. 57, pp. 232-247, 2016. https://doi.org/10.1016/j.eswa.2016.03.045

[7] S. Asarkaya, and E. Ünsal, “Recognition of Human Activities Using Machine Learning Methods on Smart Sensor Data”, Journal of Science and Technology, vol. 1, no. 1, pp. 9-14, 2022.

[8] E.A. Sağbaş, and S. Ballı, “Transportation mode detection by using smartphone sensors and machine learning”, Pamukkale University Journal of Engineering Sciences, vol. 22, no. 5, pp. 376-383, 2016. https://doi.org/10.5505/pajes.2015.63308

[9] K. Erin., C. Bayılmış, and B. Boru, “Real-Time Human Activity Detection Based on Accelerometers and Internet of Things”, Academic Platform-Journal of Engineering and Science, vol. 9, no. 1, pp. 194-198, 2021. https://doi.org/10.21541/apjes.809777

[10] R.A. Voicu, C. Dobre, L. Bajenaru, and R.I. Ciobanu, “Human physical activity recognition using smartphone sensors”, Sensors, vol. 19, no. 3, pp. 458, 2019. https://doi.org/10.3390/s19030458

[11] M. Peker, S. Ballı, and E.A. Sağbaş, “Predicting human actions using a hybrid of ReliefF feature selection and kernel-based extreme learning machine”, In Cognitive Analytics: Concepts, Methodologies, Tools, and Applications, Ed.Mehdi Khosrow-Pour , 2020, ch. 17, pp. 307-325.

[12] N. Ahmed, J.I. Rafiq, and M.R. Islam, “Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model”, Sensors, vol. 20, no. 1, pp. 317, 2020. https://doi.org/10.3390/s20010317

[13] S.W. Yahaya, A. Lotfi, and M. Mahmud, “Detecting anomaly and its sources in activities of daily living”, SN Computer Science, vol. 2, no. 1, pp. 1-18, 2021. https://doi.org/10.1007/s42979-020-00418-2

[14] R. Li, H. Li, and W. Shi, “Human activity recognition based on LPA”, Multimedia Tools and Applications, vol. 79, no. 41, pp. 31069-31086, 2020. https://doi.org/10.1007/s11042-020-09150-8

[15] M.O. Gani, T. Fayezeen, R.J. Povinelli, R.O. Smith, M. Arif, A.J. Kattan, and S.I. Ahamed, “A light weight smartphone based human activity recognition system with high accuracy”, Journal of Network and Computer Applications, vol. 141, pp. 59-72, 2019. https://doi.org/10.1016/j.jnca.2019.05.001

[16] A. Elsts, N. Twomey, R. McConville, and I. Craddock, “Energy-efficient activity recognition framework using wearable accelerometers”, Journal of Network and Computer Applications, vol. 168, pp. 102770, 2020. https://doi.org/10.1016/j.jnca.2020.102770

[17] S. K. Challa, A. Kumar, and V. B. Semwal, “A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data”, The Visual Computer, vol. 38, no. 12, pp. 4095-4109, 2022. https://doi.org/10.1007/s00371-021-02283-3

[18] İ. A. Metin, and B. Karasulu, “A novel dataset of human daily activities: Its benchmarking results for classification performance via using deep learning techniques”, Journal of the Faculty of Engineering and Architecture of Gazi University, vol. 36, no. 2, pp. 759-777, 2021. https://doi.org/10.17341/gazimmfd.772849

[19] M. Munoz-Organero, “Outlier detection in wearable sensor data for human activity recognition (HAR) based on DRNNs”, IEEE Access, vol. 7, pp. 74422-74436, 2019. https://doi.org/10.1109/ACCESS.2019.2921096

[20] B. Zhou, J. Yang, and Q. Li, “Smartphone-based activity recognition for indoor localization using a convolutional neural network”, Sensors, vol. 19, no. 3, pp. 621, 2019. https://doi.org/10.3390/s19030621

[21] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, “Deep learning models for real-time human activity recognition with smartphones”, Mobile Networks and Applications, vol. 25, no. 2, pp. 743-755, 2020. https://doi.org/10.1007/s11036-019-01445-x

[22] X. Zhou, W. Liang, I. Kevin, K. Wang, H. Wang, L.T. Yang, and Q. Jin, “Deep-learning-enhanced human activity recognition for Internet of healthcare things”, IEEE Internet of Things Journal, vol. 7, no. 7 , pp. 6429-6438, 2020. https://doi.org/10.1109/JIOT.2020.2985082

[23] M. Altuve, P. Lizarazo, and J. Villamizar, “Human activity recognition using improved complete ensemble EMD with adaptive noise and long short-term memory neural networks”, Biocybernetics and Biomedical Engineering, vol. 40, no. 3, pp. 901-909, 2020. https://doi.org/10.1016/j.bbe.2020.04.007

[24] D. Mukherjee, R. Mondal, P.K. Singh, R. Sarkar, and D. Bhattacharjee, “EnsemConvNet: a deep learning approach for human activity recognition using smartphone sensors for healthcare applications”, Multimedia Tools and Applications, vol. 79, no. 41, pp. 31663-31690, 2020. https://doi.org/10.1007/s11042-020-09537-7

[25] C. Catal, S. Tufekci, E. Pirmit, and G. Kocabag, “On the use of ensemble of classifiers for accelerometer-based activity recognition”, Applied Soft Computing, vol. 37, pp. 1018-1022, 2015. https://doi.org/10.1016/j.asoc.2015.01.025

[26] I. Elamvazuthi, L.I. Izhar, and G. Capi, “Classification of human daily activities using ensemble methods based on smartphone inertial sensors”, Sensors, vol. 18, no. 12, pp. 4132, 2018. https://doi.org/10.3390/s18124132

[27] O. Herrera-Alcántara, A.Y. Barrera-Animas, M. González-Mendoza, and F. Castro-Espinoza, “Monitoring student activities with smartwatches: On the academic performance enhancement”, Sensors, vol. 19, no. 7, pp. 1605, 2019. https://doi.org/10.3390/s19071605

[28] N. Irvine, C. Nugent, S. Zhang, H. Wang, and W.W. Ng, “Neural network ensembles for sensor-based human activity recognition within smart environments”, Sensors, vol. 20, no. 1, pp. 216, 2020. https://doi.org/10.3390/s20010216

[29] S. Brajesh, and I. Ray, “Ensemble approach for sensor-based human activity recognition”, In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, pp. 296-300, 2020. https://doi.org/10.1145/3410530.3414352

[30] R. Sekiguchi, K. Abe, T. Yokoyama, M. Kumano, and M. Kawakatsu, “Ensemble learning for human activity recognition”, In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, pp. 335-339, 2020. https://doi.org/10.1145/3410530.3414346

[31] A. Subasi, D.H. Dammas, R.D. Alghamdi, R.A. Makawi, E.A., Albiety, T. Brahimi, and A. Sarirete, “Sensor based human activity recognition using adaboost ensemble classifier”, procedia computer science, vol. 140, pp. 104-111, 2018. https://doi.org/10.1016/j.procs.2018.10.298

[32] N. Dwivedi, D.K. Singh, and D.S. Kushwaha, “Orientation invariant skeleton feature (oisf): a new feature for human activity recognition”, Multimedia Tools and Applications, vol. 79, no. 29, pp. 21037-21072, 2020. https://doi.org/10.1007/s11042-020-08902-w

[33] S. Balli, and E.A. Sağbaş, “The usage of statistical learning methods on wearable devices and a case study: activity recognition on smartwatches” Advances in statistical methodologies and their application to real problems, Ed. Tsukasa Hakimoto, 2017, ch. 13, pp. 259-277.

[34] G.H. John, P. Langley, “Estimating continuous distributions in Bayesian classifiers”, In Proceedings of the Eleventh conference on Uncertainty in artificial intelligence, pp. 338-345, August 1995.

[35] B.K. Alsberg, R. Goodacre, J.J. Rowland, and D.B. Kell, “Classification of pyrolysis mass spectra by fuzzy multivariate rule induction-comparison with regression, K-nearest neighbour, neural and decision-tree methods”, Analytica Chimica Acta, vol. 348, no. 1-3, pp. 389-407, 1997. https://doi.org/10.1016/S0003-2670(97)00064-0

[36] E.A. Sağbaş, and S. Ballı, “Estimation of Human Activities by Using Wrist Movements”, 3. International Management Information Systems Conference, pp. 52-58, October 2016.

[37] S. Ballı, E.A. Sağbaş, “Diagnosis of transportation modes on mobile phone using logistic regression classification”, IET Software, vol. 12, no. 2, pp.142-151, 2018. https://doi.org/10.1049/iet-sen.2017.0035

[38] L. Breiman, “Random forests”, Machine learning, vol. 45, no. 1, pp. 5-32, 2001. https://doi.org/10.1023/a:1010933404324

[39] S. Ballı, and E.A. Sağbaş, “Classification of Human Motions with Smartwatch Sensors”, Süleyman Demirel University Journal of Natural and Applied Sciences, vol. 21, no. 3, pp. 980-990, 2017. https://doi.org/10.19113/sdufbed.32689

[40] L. Rokach, “Ensemble-based classifiers”, Artificial intelligence review, vol. 33, no. 1, pp. 1-39, 2010. https://doi.org/10.1007/s10462-009-9124-7

[41] L. Breiman, “Bagging predictors”, Machine learning, vol. 24, no. 2, pp. 123-140, 1996. https://doi.org/10.1007/BF00058655

[42] L.I. Kuncheva, “Combining pattern classifiers: methods and algorithms”, John Wiley & Sons, 2014.

[43] E.A. Sağbaş, and S. Balli, “Activity recognition by voting method via motion sensors”, In 2017 International Conference on Computer Science and Engineering (UBMK), 2017. https://doi.org/10.1109/UBMK.2017.8093557

[44] E.A. Sağbaş, and S. Ballı, “Usage of the smartphone sensors and accessing raw sensor data”, Academic Computing Conferences, pp. 180-186, 2015.

[45] P.C. Mahalanobis, “On tests and measures of groups divergence”, IJ Asiatic Soc. Bengal, vol. 26, pp. 541, 1930.

[46] C. Leys, O. Klein, Y. Dominicy, and C. Ley, “Detecting multivariate outliers: Use a robust variant of the Mahalanobis distance”, Journal of Experimental Social Psychology, vol. 74, pp. 150-156, 2018. https://doi.org/10.1016/j.jesp.2017.09.011

[47] K. Polat, and S. Güneş, “Classification of epileptiform EEG using a hybrid system based on decision tree classifier and fast Fourier transform”, Applied Mathematics and Computation, vol. 187, no. 2, pp. 1017-1026, 2007. https://doi.org/10.1016/j.amc.2006.09.022

[48] I.H. Witten, and E. Frank, “Data mining: practical machine learning tools and techniques with Java implementations”, Acm Sigmod Record, vol. 31, no.1, pp. 76-77, 2002.