Developing an Application with Sensors in Smart Phones

Atahan Tufekci, Muhammet C. Colak, Anar Gurbanov and Pinar Kirci

Bursa UludagUniversity, Gorukle, Bursa, Turkey

Abstract
In the study, the functions of sensors in smartphone hardware, how data can be collected from sensors during the coding phase, and healthy driving, fainting detection applications will be developed and the results will be discussed.

Keywords
sensors, smartphone, driving

1. Introduction

With the advances and developments in smartphones, various sensors such as accelerometers, gyroscopes, magnetometers and similar sensors have been included in the hardware of these devices. Thanks to these sensors, it is possible to obtain various data about the person using the device and its environment. These data can be accessed with the libraries used in the mobile software development phase. Applications were developed by processing these data accessed with mathematical methods or machine learning algorithms. With these applications, it has been possible to obtain results in many areas about people using smartphones.

In the age of technology we live in, smartphones are perhaps the devices that people use, spend time with and keep with them the most. At first glance, smartphones are thought to be used only for communication and communication, but with the advancement of technology, new methods and new directions have been discovered in which these devices can be used. Over time, with the development of the hardware of smartphones, sensors such as accelerometers and gyroscopes (1) have been included in the hardware of these devices and new usage opportunities have emerged for smartphones. With the help of these sensors, it is possible to obtain various information about the person using the phone and the changes in their environment. By analyzing the data collected from the activities and movements of the person during the day with mathematical calculations or machine learning algorithms, results have emerged in the field of health, where people can detect and track their own fainting, driving, stepping, running, stopping, sitting (2). It has been possible to develop applications related to this subject.

2. Sensors used in the project

2.1. Accelerometer Sensor

The accelerometer sensor in the smartphone hardware is used to measure the acceleration applied to the device. The data obtained from this sensor gives the acceleration value affecting the smartphone in the x, y, z axes in plus and minus directions. These values are in g (1). Based on these data, the movements made by the person using the smartphone and the shaking of the device analyzes can be made regarding their states and vibrations (3). Based on these analyzes, certain inferences can be made. In the coding phase, the variables and functions of the Accelerometer sensor type and SensorEventListener interface from the SensorManager library (1) can be used in the Android system to access the data collected by this sensor. In this way, accelerometer data can be obtained with

COLINS-2024: 8th International Conference on Computational Linguistics and Intelligent Systems, April 12–13, 2024, Lviv, Ukraine
pinarkirci@uludag.edu.tr (P. Kirci);
0000-0002-0442-0235 (P. Kirci);

smartphones that people carry with them all day long without the need to use an extra environment, device or sensor (4).

2.2.Gyroscope Sensor

The gyroscope sensor in the smartphone hardware is used to measure the orientation angle of the device in the x, y, z axes in the plus and minus directions (5). With the data obtained from this sensor, the orientation state, rotation angle and angular velocity of the smartphone can be determined (3). By using the data collected from the gyroscope sensor, certain inferences can be made from the analysis of the physical activities of the person using the smartphone in daily life, as used in the accelerometer sensor. In order to access the data collected by this sensor during the coding phase, the variables and functions of the Gyroscope sensor type and SensorEventListener interface from the SensorManager library (1) can be used in the Android system. In this way, gyroscope data can be obtained from smartphones that people often carry with them without the need for an auxiliary factor.

3. Areas of Use of Sensors

3.1. Step Detector

One of the most important aspects of personal health is to stay active during the day. It is necessary for a healthy life for people to get out of situations where they remain stationary and move, walk and run. Thanks to the studies on the step detector, people have had the opportunity to track how many steps they take during the day. Although different tools can be used for step detection and step counting, accelerometer sensors are generally used. For this process, basically 4 stages can be mentioned as data acquisition from the accelerometer sensor, noise reduction, detection of stepping and whether to count it as a step or not (6). Although the most common method is to calculate the magnitude of the x, y, z data from the accelerometer sensor and compare this value with a threshold value, many different methods have also been used. In some studies, the x, y, z data from the accelerometer sensor were sent into a mathematical function and the output value was used together with the zero crossing method (6) (7). In some studies, the user's walking frequency was also included in the calculations (8). In addition to the different methods used, studies have also taken into account the activity in which the data from the accelerometer was obtained while the person was performing the activity or the position in which the smartphone was in during the determination of stepping. In a previous example study on this topic, the values of the three axes obtained from the accelerometer sensor were processed with various formulas and a net magnitude value was calculated. In the next step, a dynamically running peak detection algorithm and an algorithm for detecting false peaks and real peaks were used using this calculated magnitude value. In the following steps, step detection was performed using the start vector and end vector values obtained by these algorithms. Using the data obtained with the algorithms and methods used, the step length was determined and the distance traveled was estimated accordingly (9).

In another study on step detector and step detection, values such as sampling rate, stepping rate and average of orthogonal accelerations are taken into account while determining the threshold value.

The algorithm was designed and tested at different walking speeds and different phone positions. As a result of the tests, very high success percentages were obtained (10).

3.2. Vehicle Use Behaviors

Although it is not directly related to human health, both physical and psychological health of a person can be injured in a traffic accident that may occur. The majority of these traffic accidents are caused by human errors and the effects of these errors. In order to prevent or minimize the negativities that may occur as a result of traffic accidents, it has been possible to make applications using the sensors in the smartphone hardware. In these applications, two issues related to driving behavior can be examined: the behavior of the driver using the vehicle and the way the vehicle is used (11). The driver's behavior can include whether the driver is focused on the road or whether the driver is on the phone while driving. The manner in which the vehicle is driven can include aggressively pressing the gas and brakes or making very sharp turns on bends. In a study on this topic, the data obtained from the accelerometer and gyroscope sensors on the smartphone in the vehicle were used, and the driver's behaviors such as aggressive cornering, aggressive brake pedal use and aggressive accelerator pedal use were detected by comparing the sensor data using the DTW algorithm. With a certain number of signals, the driver's maneuvers were identified (12). When another study was examined, it was seen that classes such as aggressive driving, normal driving, aggressive braking and normal braking were determined in order to detect driving behaviors with the accelerometer sensor and datasets of the data obtained from the accelerometer sensor related to these classes were created. In the following stages, classification processes were performed and tested with the help of MLP (Multilayer Perceptron), RF (Random Forest), KNN (K-Nearest Neighbors) and GNB (Gaussian Naïve Bayes) algorithms and methods.

After the tests, F-scores for the classification of different event types were observed (13). Various studies have been conducted with different classifications, methodologies and methods, such as using the ANN (Artificial Neural Network) algorithm (14). It has been observed that various success percentages have been observed with different classification algorithms and different scenario classes.

3.3. Activity Detection

Another topic related to the field of smartphone sensors is activity detection. With the accelerometer and gyroscope sensors detecting information such as vibration, rotation, acceleration, deceleration, tilt, etc., the orientation of smartphone users' activities such as running, walking and climbing can be determined. Thanks to the sensors of systems such as smartphones, it has been possible to receive data, process the data and perform motion detection without the need to integrate another external sensor (15). For activity detection, data is first collected from mobile sensors such as accelerometers and gyroscopes during different activities. After collecting the data, certain features such as speed and orientation are extracted for each activity. After these features are extracted, tests are performed and the activity is detected according to the result of the test. For activity detection, machine learning and classification algorithms such as kNN (2), Logistic Regression (2), SVM, Random Forest, Naive Bayes, Bayesian Networks, Multilayer Neural Network, Ameva, K-Means Clustering (16) are important. In a study on this subject, activity detection was performed by classifying with SVM, LR and J48 decision tree algorithms. Activities were used as standing, walking, running, running, lying down, getting up, and getting down.

As a feature, mean, energy, entropy, standard deviation and correlation were used. Accelerometer and gyroscope sensors were used to measure and collect the data. The highest percentage results for all three algorithms were obtained for the lying down activity. The lowest result was recorded in the detection of sitting activity with the SVM algorithm. In general, when looking at the average of all, the LR algorithm gave the highest result (17). (18) compared the same six activities with SVM and HF-SVM (Hardware Friendly SVM). 789 test instances were evaluated with approximately equal instances per class and it was observed that although the percentage of detection of up and down activities was low in both algorithms, the overall detection percentage was high. In the study by (19), XGB, SVM, NN, Soft voting, Hard voting algorithms and gyroscope sensor were used. 2 different numbers of features (195 and 304) were tested. Normal walking, fast walking, going down, going up, going down, going up and sitting were used as activities. Testing was done for three different situations. In the first case, bag walking and normal walking were combined as a single activity. In the second case, normal, fast and bag walking were combined as a single activity. The results showed that the Soft Voiting approach had the highest detection results.

3.4. Condition Monitoring in Parkinson's Disease

Among the many studies conducted in the field of health with integrated sensors used in smartphones, one of the most important ones is the monitoring of the condition of Parkinson's patients. Using accelerometer and gyroscope sensors in smartphone hardware, measurements such as tremor, slow movement and loss of balance can be used to evaluate motor functions and diagnose as well as severity (20). Neural networks including back-propagation algorithms are frequently used in the problem of classifying data about patients and in solving this problem. In addition, Levenberg - Markard algorithm and conjugate gradient algorithms have also been used (21). In a study on this subject, during the data analysis and preprocessing process using Python, the sensor data of each participant were organized by side (right or left), sensor mode, activity and session, and then organized according to the timestamp with the BioStampRC application. Left (right) side sensor data were matched to the corresponding side clinical scores for bradykinesia, tremor and dyskinesia, and all sensor data obtained from the accelerometer and gyroscope were segmented into 5-second clips with 50% overlap. To remove the limb orientation effect and to detect bradykinesia, high or low pass filters were applied to the accelerometer and gyroscope data at specific frequencies. These filters were found to be helpful in symptom detection. Each of the accelerometer and gyroscope data from a total of 41,802 clips obtained in this process was matched with the corresponding patient ID, side, activity, session and clinical score data (22).

4. Project Presented

4.1. Healthy Driving Practice

Within the scope of healthy driving, our application aims to focus the driver on the road and driving in situations where the driver is talking on the phone or interested in the phone. In this application, 3 classes have been determined and these classes include the state of the phone in the holder / in front of the user while driving, the state of the phone in the pocket while driving, and the state of talking on the phone while driving (23). In order to perform the classification process, KNN (K-Nearest Neighbors) algorithm (13) was used with a dataset and classification was performed with different K values such as 3, 5, 7, 9, 11. The success of the classification process performed with different K values in the KNN algorithm.

The classified scenarios were realized with the application and our testing processes were carried out in order to determine its status. For the data collection process to be used in the classification process of our application, sample scenarios belonging to 3 classes were realized through the application we wrote with Android Studio and Java, and the x, y, z axis data obtained from the accelerometer sensor with a frequency of 0.2 Hz were collected and saved in a file to create our data set in the specified format. Figures 1, 2 and 3 below show the graphs of the accelerometer data collected for the sample scenarios that constitute our dataset.

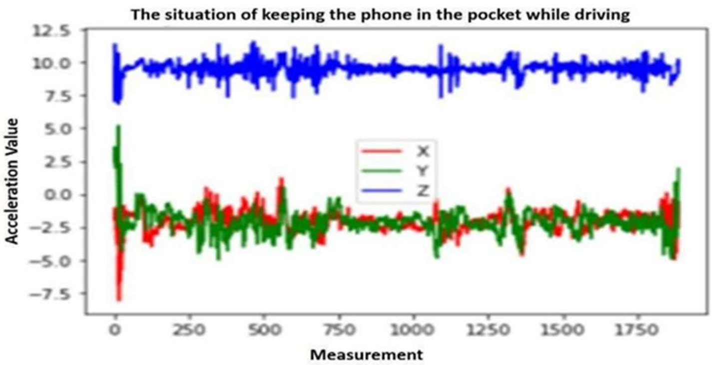

Figure 1: Accelerometer data collected with the driver's phone in his pocket

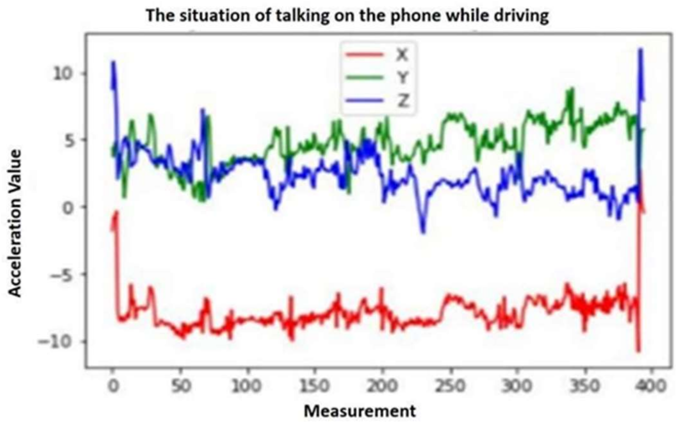

Figure 2: Accelerometer data collected while the driver was talking on the phone

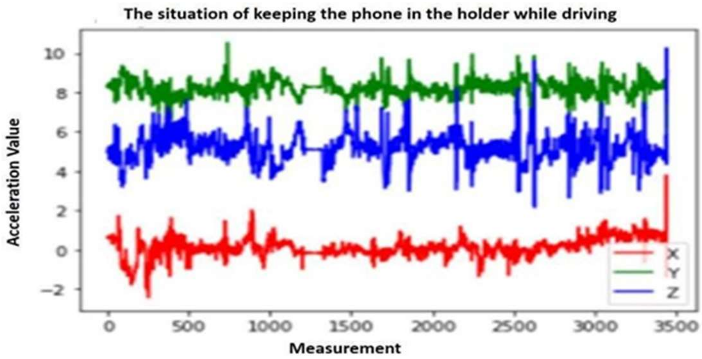

Figure 3: Accelerometer data collected while the phone is in the holder

4.2. Fainting Detection App

Within the scope of fainting detection of the application made by us, it is aimed to send notifications to the relevant people in cases where we detect the user fainting. In this application, 4 classes have been determined and these include the user's sitting state, the user's walking state, the user's standing state and the user's fainting state (24). In order to perform the classification process, KNN (K-Nearest Neighbors) algorithm (10) was used with a dataset and classification was performed with different K values such as 3, 5, 7, 9, 11. In order to determine the success of the classification process performed with different K values in the KNN algorithm, the classified scenarios were realized with the application and our testing processes were carried out.

For the data collection process to be used in the classification process of our application, we realized sample scenarios of 4 classes through the application we wrote with Android Studio and Java and collected data from the accelerometer and gyroscope sensor.

With a frequency of 0.2 Hz, the x, y, z axis data were collected and saved in a file with xls extension in the specified format to form our data set. Figures 4 and 5 below show the graphs of the accelerometer and gyroscope data collected for the sample fainting scenario that constitutes our dataset.

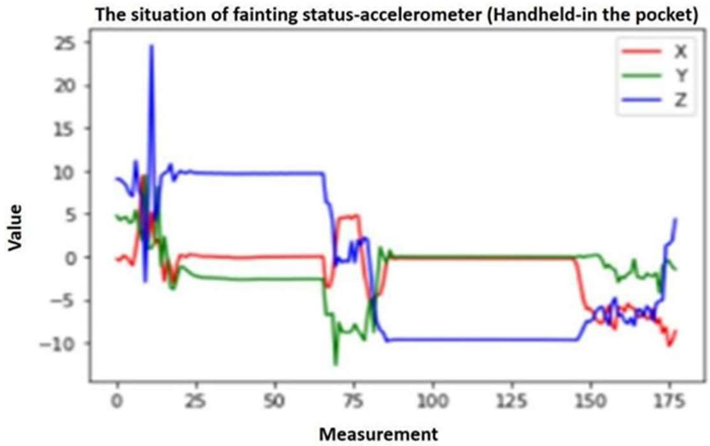

Figure 4: Accelerometer data for the sample fainting scenario

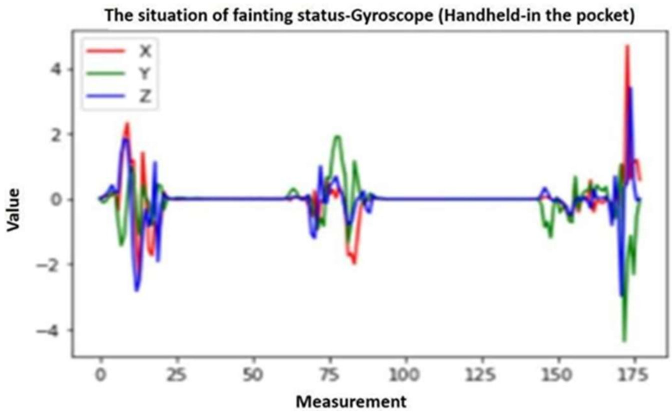

Figure 5: Gyroscope data for the example fainting scenario

4.3. Step Counter App

Within the scope of the step counter of the application made by us, it is aimed to detect the user's stepping motion with step detection methods and display it with a counter and accordingly display information such as distance, speed, energy expenditure.

Each time the data coming from the accelerometer with a certain frequency changes, we get a result from the functions of formula (4.1) and formula (4.2) given below (7).

$$ivme = \cot \frac{y}{z} \qquad (4.1)$$

$$ivme = \cot \frac{x}{\sqrt{y^2+z^2}} \qquad (4.2)$$

The step will be counted when we detect that the result obtained from the functions has transitioned from negative to positive or from positive to negative using the zero crossing method. In order to eliminate the problems that may arise from the detection sensitivity of the accelerometer sensor (9), the validity of a step is determined by the requirement of a minimum of 0.25 seconds more than the previous valid step (7).

On the one hand, while the number of steps is determined, the distance information is calculated using the height information and the number of steps we receive from the user, the average speed information is calculated using the distance and time change, and the amount of energy consumed is calculated using the weight information we receive from the user (25). We calculate the average speed information with the formula (4.3) given below.

$$hız = \frac{\Delta x}{\Delta t} \quad (4.3)$$

5. Research Results

5.1. Step Counter App Results

In the step detection part, which is the basis of our step counter application, the x, y, z values obtained from the accelerometer sensor were calculated with the formulas and zero crossing method we have previously mentioned (7). In order to determine the accuracy of our operations, we tested the success of the step detector application in cases where the user holds the phone steady in his/her hand, the user shakes the phone in his/her hand, and the user's phone is in the pocket (10) and the following results were obtained.

Success percentage of the step detection algorithm:

Phone held steady in hand, 99.0%

Phone shaken in hand, 87.0%

Phone in pocket, 93.0%

In addition to step detection, the user can track and monitor the values of time, distance, speed andenergy expenditure with the information we receive from the user and other calculations we have made.

5.2. Fainting Detection Implementation Results

In our fainting detection application, we performed classification using the KNN (KNearest Neighbor) algorithm (10) on the data we obtained. In order to find the appropriate K value to be used in the KNN algorithm, the classification success of different values were tested and our specific results were obtained. With the data obtained by the accelerometer sensor on the smartphone, the classification process at a frequency of 1 Hz was performed with the algorithm we coded and analyzed by us by viewing it on the application. Classification processes and classified scenarios of the user's sitting state, the user's standing state, the user's walking state and the user's fainting state (24), which are the 4 classes we have determined, were performed with the application and our testing processes were carried out.

The results we have obtained are shown in our success percentage graph displayed in Figure 6 below with the data visualization methods we have made through Python.

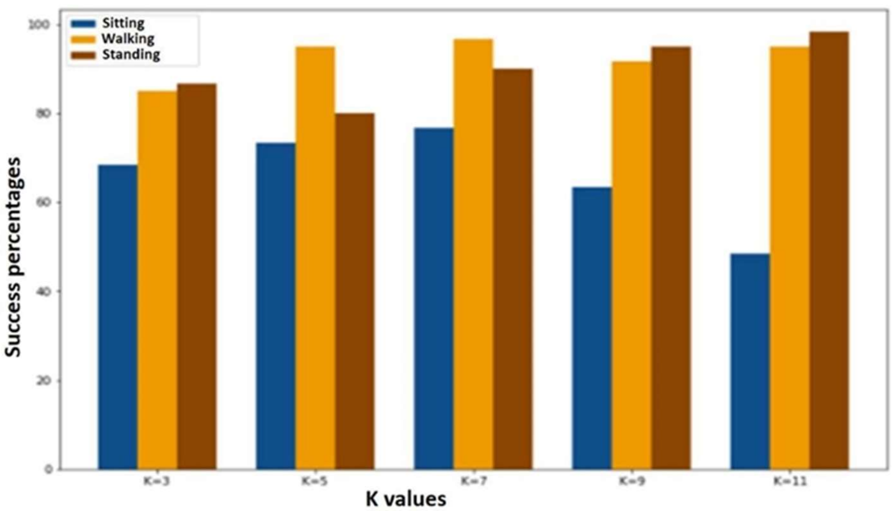

Figure 6: Classification success percentages in our fainting detection application

As a result of our tests, in order to determine the K value to be used in the KNN algorithm, the average classification success of each K value was examined by us. As a result of our examination, we found that when K is 3, there is an average success of 71.25%, when K is 5, there is an average success of 76.67%, when K is 7, there is an average success of 77.92%, when K is 9, there is an average success of 72.92%, when K is 11, there is an average success of 65.42%, and based on the results of our observations, the value to be used in the algorithm was determined.

As a result of the data we collected and the classification we made with the KNN algorithm, if the user is detected to be unconscious, an application that sends an SMS notification to a specific phone number was written by us using Android Studio and Java.

5.3. Healthy Driving Practice Results

In our healthy driving application, the classification process was performed using the KNN (K-Nearest Neighbor) algorithm (13) on the data we obtained.

KNN in order to find the appropriate K value to be used in the algorithm, the classification success of different values was tested and our specific results were obtained. With the data obtained by the accelerometer sensor on the smartphone, the classification process at a frequency of 1 Hz was performed with the algorithm we coded and analyzed by us by displaying it on the application.

Classification processes and classified scenarios for the 3 classes we have determined, which are the state of having the phone in the holder/front while driving, the state of having the phone in the pocket while driving, and talking on the phone while driving (23), were carried out together with the application and our testing processes were carried out.

The results obtained are shown in our success percentage graph displayed in Figure 7 below with the data visualization methods we made through Python.

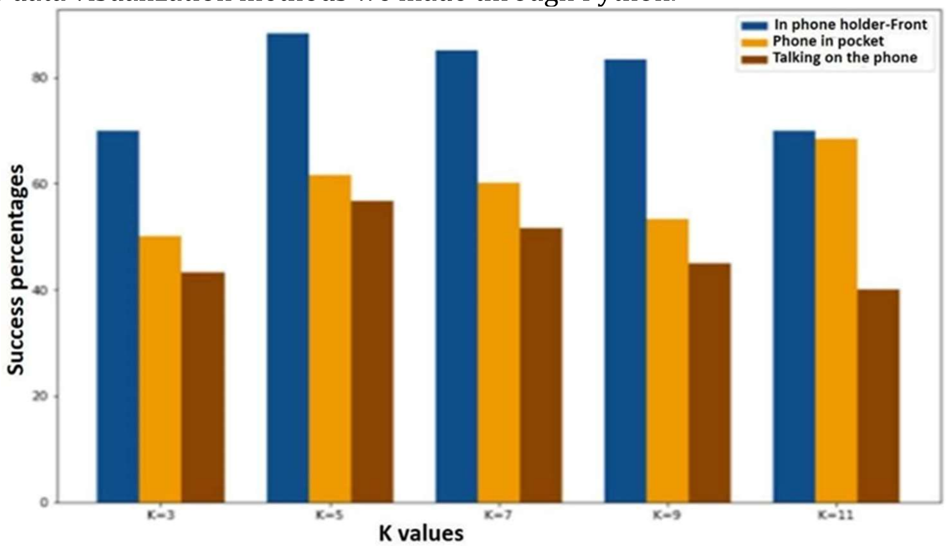

Figure 7: Classification success percentages in healthy driving

As a result of our tests, in order to determine the K value to be used in the KNN algorithm, the average classification success of each K value was examined by us. As a result of the examination, we found that when K is 3, the average success rate is 54.44%, when K is 5, the average success rate is 68.89%, when K is 7, the average success rate is 65.56%, when K is 9, the average success rate is 60.55%, when K is 11, the average success rate is 59.44% and based on the results of our observations, the value to be used in the algorithm was determined. As a result of our data set and the classification process we performed with the KNN algorithm, the actions to be taken and the actions to be taken depending on the user's use of the smartphone while driving are provided by the mobile application written by us with Android Studio and Java.

5.4. Application Overview

Figure 8 below shows the interface of the step counter application, where the number of steps taken by the users is determined and the users can observe various information as a result of the other calculations we have made. When users specify their weight and height and open the application, they will be able to access the relevant information on this interface.

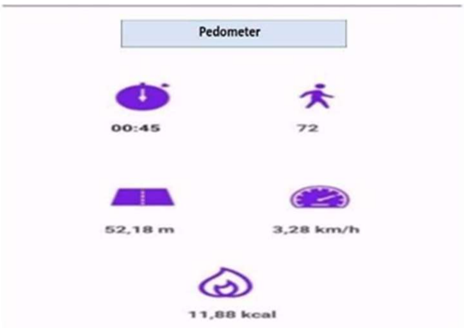

Figure 8: Interface of our step detection application

Figure 9 below shows the KNN algorithm we used, the classes we determined and the data sets we obtained as a result of the data collection process we carried out; There is an interface that allows us to observe how much success percentage we have achieved in the classification process, which is the basis of our application.

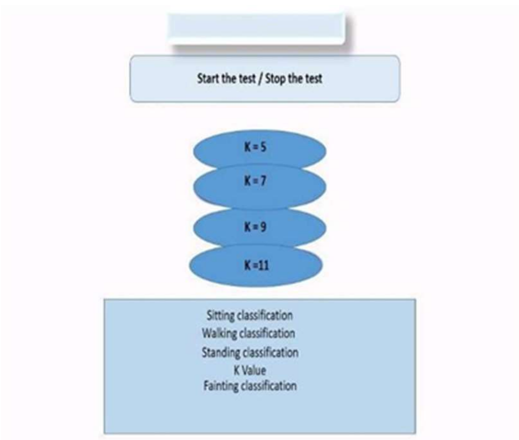

Figure 9: Fainting detection classification interface

An audible warning is given to indicate that people using our application are not focusing on the road during driving activity. Again, when the fainting condition of the users of our application is detected, a notification is sent to the mobile phone number specified in the application. Figures 10 and 11 below show the interfaces where information appears on the screen according to the classifications made while the application is running.

Figure 10: Fainting detection application information interface

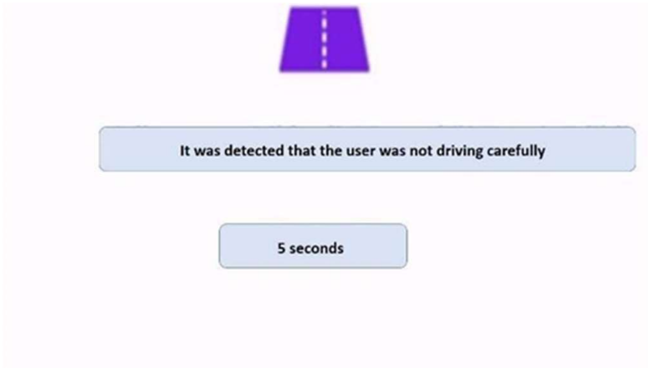

Figure 11: Healthy driving application information interface

6. Conclusion

In our study, different procedures in previous studies or information obtained from other sources have produced different results. The zero-crossing method and formulas (7), which we used as valid, obtained more successful and useful results than statically determining a limit value and comparing the magnitudes of the axis values obtained from the accelerometer sensor with the limit value and in the zero-crossing method, the measurement accuracy of the accelerometer sensor is not paid attention.

The average success percentages of previous studies and our study are shown below for comparison and evaluation purposes.

Average success percentages of step detector studies: Average Success in our study is 93.00%, Average Success in the study of (7) is 97.09%, Average Success in the study of (10) is 97.97%.

Although the main purpose of our application is a step counter, we have created an application where the user can observe the distance traveled, the average speed and the energy consumed, along with the calculations we have made. Thus, it is possible to observe different features that users may want to know.

In the general scope of our fainting detection application, we believe that detecting the fainting of the person using the smartphone, sending a notification with this detection and tracking the other activities of the user is useful for people who need or may need help in these matters. In the general scope of our healthy driving application, we believe that it would be useful to detect and warn the user if the person using the smartphone is interested in the phone or talking on the phone while driving.

The KNN (K-Nearest Neighbor) algorithm was used for both applications in the classification sections of our application. With the use of this algorithm, we tested the success of classification scenarios with K values in a certain range (3, 5, 7, 9, 11). When we examined the success percentages after our tests, we observed that some of the different K values and different classes had a high success percentage, some had an average success percentage, and some had a low success percentage.

Below, the highest success percentages of the classifications made in our faint detection and healthy driving applications in different studies and in our study are given for comparison and evaluation purposes.

In our study, healthy driving had the highest classification success: Phone in Holder/Front 88.33%, Phone in Pocket 68.33%, Talking on the Phone 56.67%.

Fainting detection application highest classification success:

In our study; 76.67% in sitting, 96.67% in walking, 98.33% in standing, 58.33% in fainting.

In the study of (17); 94.2% in the Sitting state, 97.2% in the Walking state, 94.8% in the Standing state, - in the Fainting state.

In the study of (18); 96.4% in the Sitting state, 95.6% in the Walking state, 93.0% in the Standing state, in the Fainting state.

References

[1] E. Sagbas, S. Ballı, The Use of Smartphone Sensors and Access to Raw Sensor Data, in : XVII. Akademik Bilisim Konferansi, AB 2015, 4-6 february 2015

[2] E. Sagbas, S. Ballı, Comparison of Logistic Regression and kNN Methods in ActionRecognition with Smartphone Sensor Data, in: 1st International Conference on Engineering Technology and Applied Science 2016, pp 21-22

[3] R. Kusuma, S. Mukheerjee, Health Monitoring with Smartphone Sensors and Machine Learning Techniques, in: 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2023, pp. 576-581, doi: 10.1109/ICAAIC56838.2023.10140210.

[4] Aydemir, E., Karslioglu, I. "Gait Pattern Analysis with the Help of Accelerometer Sensor of Smartphones." Afyon Kocatepe University Journal of Science and Engineering Sciences, 21(2), 283-299. https://doi.org/10.35414/akufemubid.856995

[5] Alruban Abdulrahman, Al-obaidi Hind, Clarke Nathan, Li Fudong, " Physical Activity Recognition by Utilizing Smartphone Sensor Signals." 10.5220/0007271903420351, (2019).

[6] Seo Jungryul, Chiang Yutsai, Laine Teemu, Khan Adil, "Step counting on smartphonesusing advanced zero-crossing and linear regression." ACM IMCOM 2015 - Proceedings.10.1145/2701126.2701223.

[7] S. Y. Park, S. J. Heo and C. G. Park, Accelerometer-based smartphone step detection using machine learning technique, in: 2017 International Electrical Engineering Congress (iEECON), Pattaya, Thailand, 2017, pp. 1-4, doi: 10.1109/IEECON.2017.8075875.

[8] Kang Xiaomin, Huang Baoqi, Qi Guodong, A Novel Walking Detection and Step Counting Algorithm Using Unconstrained Smartphones. Sensors (Basel, Switzerland). 18. 10.3390/s18010297, 2018.

[9] Alabadleh Ahmad, Hawari Eshraq, Alkafaween Esra'a, Alsawalqah Hamad, Step Detection Algorithm For Accurate Distance Estimation Using Dynamic Step Length. 324-327. 10.1109/MDM.2017.52, 2017.

[10] H. Muhsen, O. Al-Amaydeh, R. Al-Hamlan, "Algorithm Design for Accurate Steps Counting Based on Smartphone Sensors for Indoor Applications", Advances in Science, Technology and Engineering Systems Journal, vol. 5, no. 6, pp. 811-816, 2020.

[11] Mantouka Eleni, Barmpounakis Manos, Vlahogianni Eleni, Golias John, "Smartphone Sensing for Understanding Driving Behavior: Current Practice and Challenges." International Journal of Transportation Science and Technology. 10. 10.1016/j.ijtst.2020.07.001, 2020.

[12] D. A. Johnson and M. M. Trivedi, Driving style recognition using a smartphone as a sensor platform, in: 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC),Washington, DC, USA, 2011, pp. 1609-1615, doi: 10.1109/ITSC.2011.6083078.

[13] M. R. Carlos, L. C. González, J. Wahlstrom, G. Ramírez, F. Martínez and G. Runger, How Smartphone Accelerometers Reveal Aggressive Driving Behavior?-The Key is the Representation, in: IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 8, pp. 3377-3387, Aug. 2020, doi: 10.1109/TITS.2019.2926639.

[14] S. K. Sonbhadra, S. Agarwal, M. Syafrullah and K. Adiyarta, Aggressive driving behaviour classification using smartphone's accelerometer sensor, in: 2020 7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI), Yogyakarta, Indonesia, 2020, pp.77-82, doi: 10.23919/EECSI50503.2020.9251913.

[15] K. Erin, C. Bayilmis, and B. BORU, "Accelerometer and Internet of Things Based Real-Time Human Activity Detection", APJES, vol. 9, sy. 1, pp. 194-198, 2021, doi: 10.21541/apjes.809777

[16] Iskanderov, J., Guvensan, M. A. "Activity recognition with smartphones and wearables: Classical approaches, new solutions." Pamukkale University Journal of Engineering Sciences, 25(2), 223-239, 2019.

[17] W. C. Hung, F. Shen, Y.L. Wu, M. -K. Hor and C. Y. Tang, Activity Recognition with sensors on mobile devices, in: 2014 International Conference on Machine Learning and Cybernetics, Lanzhou,China, 2014, pp. 449-454, doi: 10.1109/ICMLC.2014.7009650

[18] Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J.L. Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine, in: Bravo, J., Hervás, R.,Rodríguez, M. (eds) Ambient Assisted Living and Home Care. IWAAL 2012. Lecture Notes in

Computer Science, vol 7657. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35395-6_30, 2012.

[19] Alruban Abdulrahman, Al-obaidi Hind, Clarke Nathan, Li Fudong, “ Physical ActivityRecognition by Utilizing Smartphone Sensor Signals.“, 10.5220/0007271903420351, 2019.

[20] F. Onay, B. Karacali, Accelerometer-Based Timing Analysis for Parkinson's Disease Classification, in: 2023 31st Signal Processing and Communications Applications Conference (SIU),Istanbul, Turkey, 2023, pp. 1-4, doi: 10.1109/SIU59756.2023.10223916

[21] Shichkina Yulia, Kataeva Galina, Yulia Irishina, Elizaveta Stanevich. “The Architecture of the System for Monitoring the Status in Patients with Parkinson's Disease Using Mobile Technologies.”10.1007/978-3-030-32258-8_62, 2020.

[22] Lonini, L., Dai, A., Shawen, N. et al.” Wearable sensors for Parkinson's disease: which data are worth collecting for training symptom detection models.” npj Digital Med 1, 64 (2018). https://doi.org/10.1038/s41746-018-0071-z

[23] Fu Li, Hai Zhang, Huan Che and Xiaochen Qiu, Dangerous driving behavior detection using smartphone sensors, in: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, 2016, pp. 1902-1907, doi: 10.1109/ITSC.2016.7795864.

[24] A. Ferrari, D. Micucci, M. Mobilio and P. Napoletano, "On the Personalization of Classification Models for Human Activity Recognition," IEEE Access, vol. 8, pp. 32066-32079, 2020, doi: 10.1109/ACCESS.2020.2973425.

[25] Okmayura Finanta, Effendi Noverta, Ramadhani Witri, Jefiza Adlian, (2019). Analysis and Design of Calories Burning Calculation in Jogging Using Thresholding Based Accelerometer Sensor. 10.2991/iccelst-st-19.2019.3, 2019.