Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 3094 – 3110

doi:10.3906/elk-1910-139

Exploring the parameter space of human activity recognition with mobile devices

Berrenur SAYLAM[1], **Muhammad SHOAIB**[2], **Özlem DURMAZ İNCEL**[3,*]
[1]Deparment of Computer Engineering, Faculty of Engineering, Boğaziçi University, İstanbul, Turkey
[2]Pervasive Systems Research Group, EEMCS Faculty, University of Twente, Enschede, the Netherlands
[3]Deparment of Computer Engineering, Faculty of Engineering and Technology,
Galatasaray University, İstanbul, Turkey

Received: 21.10.2019 • **Accepted/Published Online:** 04.05.2020 • **Final Version:** 30.11.2020

Abstract: Motion sensors available on smart phones make it possible to recognize human activities. Accelerometer, gyroscope, magnetometer, and their various combinations are used to classify, particularly, locomotion activities, ranging from walking to biking. In most of the studies, the focus is on the collection of data and on the analysis of the impact of different parameters on the recognition performance. The parameter space includes the types of sensors used, features, classification algorithms, and position/orientation of the mobile device. In most of the studies, the impact of some of these parameters is partially analyzed; however, in this work, we investigate the parameter space in detail with a global focus. Particularly, we investigate the impact of using different feature-sets, the impact of using different sensors individually and in combination, the impact of different classifiers, and the impact of phone position. Using an ANOVA analysis, we investigate the importance of various parameters on the recognition performance. We show that these parameters are ranked according to their impact on the recognition performance in the following order: sensor, position, classifier, feature. We believe that such an analysis is important since we can statistically show how much a parameter is affecting the recognition performance. Our observations can be used in future studies by only focusing on the important parameters. We present our findings as a discussion to guide the further studies in this domain.

Key words: Human activity recognition, motion sensors, machine learning, wearable computing

1. Introduction

Human activity recognition using motion sensors available on mobile devices has been investigated in various studies [1]. Often, a study focuses on the collection of data and analyzing the impact of different parameters on the recognition performance. Parameter space comprises the types of sensors used, whether the sensor data are fused or used separately, data preprocessing methods, features, classification algorithms, and position/orientation of the mobile device. In most of the studies, the impact of some of these parameters is analyzed. For example, in [2], the focus is on analyzing the impact of sensor fusion while in [3], impact of different features on the recognition performance is considered.

In this paper, our aim is to study the parameter space in detail. Particularly, we investigate the impact of using different feature-sets, impact of using different sensors individually and in combination, impact of different classifiers and impact of a phone's position. While in our preliminary work [2], we analyze the impact of sensor fusion using different sets of features, in this paper, we expand the feature-set and include 15 different features, extracted from X, Y, Z-axis readings of each sensor as well as the magnitude value computed from

*Correspondence: odincel@gsu.edu.tr

these 3-axes. In total, 60 features are extracted from each sensor except pitch and roll that are calculated for accelerometer or linear acceleration sensors.

For our analysis, we use a dataset[1] [2], which includes ten participants performing seven activities. The dataset includes readings from four sensors: accelerometer, gyroscope, magnetometer, and linear acceleration. Participants carried five phones at different positions. We start with extracting different sets of features. Features include the used sets of features in similar studies presented in a survey in [3]. In the next phase of our analysis, we look at the performance of recognition with different types of sensors. We observe that, when sensors are used individually, accelerometer performs the best. When they are combined, accelerometer, gyroscope, and magnetometer combination achieves the highest recognition performance. If we compare the performance of classifiers, random forest and support vector machine (SVM) perform better than decision tree and Gaussian naive Bayes. Average performance with the position-specific analysis is observed to be between 70% and 90% depending on the position and feature-set when a single sensor is used. When sensors are combined, up to 97% performance is achieved.

After exploring the feature and sensor parameter space in detail, we apply an ANOVA analysis to investigate the impact of each parameter on the results, rather than only making observations and comparisons about the results. Although in [4], ANOVA analysis is used, it mainly focuses on the effect of sampling rate and window size using RF classifier. Our study is among the first to investigate the impact of each parameter, different from partially selected features, using an ANOVA analysis among the studies that focus on human activity recognition using motion sensors available on mobile devices. We believe that such an analysis is important since we can statistically show how much a parameter is affecting the recognition performance. Although previous studies also analyze the impact of each parameter on the recognition scores, they only show which parameter setting performs better/worse than another. For example, in [2], we show which sensor or sensor combination performs better, but we do not show how much the sensor type affects the recognition result. Since the parameter space is large, including 11 sensor combinations, 7 feature-sets, 4 classifiers, and 5 positions, it is difficult to interpret the results. Hence, ANOVA analysis makes it easy to discover the impact of each parameter. ANOVA results show that all parameters are significant on the performance recognition and the ranking is as follows: i) sensor, ii) position, iii) classifier, iv) feature.

Overall, our main contribution is the exploration of the whole parameter space where we gather results with all possible combinations of these parameters and show the most important parameters on the recognition performance. These results can guide new studies that focus on human activity recognition with mobile devices. The main highlights of the paper are as follows:

- We perform a detailed analysis with features. We investigate the use of a large number of features (in total 300 features: 15 feature types for all axis of four sensors), and different feature-sets. Besides a large set of features, we explore a parameter space including 11 sensor combinations, 7 feature-sets, 4 classifiers, and 5 positions.
- We present an ANOVA analysis showing the impact of different parameters on the recognition success: sensors, classifiers, positions, and activities.
- Our results show that, while all parameters are significantly important, sensors affect the performance much more than the other parameters.

[1] Activity Recognition Dataset (2013). University of Twente [online]. Website https://www.utwente.nl/en/eemcs/ps/dataset-folder/activity-recognition-dataset-shoaib.rar [accessed 10 June 2020]

The rest of the paper is organized as follows: in Section 2, we explain the previous studies and how this study differs from those. In Section 3, we explain our methodology. In Section 4, we analyze the impact of each parameter on the recognition performance together with the ANOVA analysis and also in Section 4.5, we summarize our findings and discuss how these can be improved. Finally, in Section 5, we conclude the paper.

2. Related work

The performance of a human activity recognition process depends on various factors such as sensors, body positions, feature-sets, classifiers, and the set of activities being recognized. The impact of these individual aspects has been extensively studied in the last few years [2]. However, this has been done mainly in a limited context, i.e. on an individual level and not in comparison with each other.

For example, in some cases, only one accelerometer is used [1, 2, 5, 6]. In other cases, different sensors such as gyroscope and magnetometer are combined with an accelerometer, but their individual contributions are not evaluated in detail [7–10]. In [11], the authors show no improvements in the overall recognition performance of various activities due to the addition of gyroscope with the accelerometer. They used naive Bayes, decision tree, and K-nearest neighbor for classification whereas multiple body positions was used in this study. In [12], a combination of the accelerometer and gyroscope was also used and found the gyroscope to be useful. However, they used only KNN classifier involving only pocket position. In [13], the authors used a combination of multiple sensors (accelerometer, magnetometer, gyroscope, linear acceleration, gravity) and reported improved results. However, the paper did not discuss the role of the individual sensors. Therefore, it is not clear which sensors contributed (and how much) to the improvement in the activity recognition.

Though there are individual studies on different aspects of a human activity recognition process, there is a need for a comprehensive study which covers all these aspects in similar experimental setup. For this, we initially investigated the role of smart phone sensors in detail in [2]. However, we did not evaluate the impact of other aspects, features, positions, etc., on the recognition performance compared to others. In this study, we address these issues and improve upon the existing work with a more detailed analysis. We also use ANOVA analysis to investigate the effect of these individual aspects of a human activity recognition process on the overall recognition performance. We use a more extensive feature-set which were analyzed in different studies in the literature [3] not together.

In the literature, it is known that ensemble methods, such as random forest, have higher performance rather than classical classification methods such as SVM because of their "perturb and combine" strategies [14]. Furthermore, there exist studies which utilize more complex classification methods, such as Bayesian network-based probabilistic generative framework [15] and neural networks [16]. However, in this work, we want to combine the most commonly used methods, such as SVM, decision tree, random forest, and Gaussian naive Bayes to have a global understanding in terms of feature-sets, sensors, positions, and activities. Additionally, deep learning techniques [17] allow the feature extraction and selection automatically and the evaluation of feature-sets is not required to be performed manually. One of the main reasons for not using deep learning methods in this work is that our dataset size is not large enough to apply a deep learning technique. Another reason is that a trained linear model has weights which are interpretable and give useful information about the activities to be recognized. On the other hand, training a convolutional neural net is very slow and tuning of the hyperparameters makes it even slower. Considering the resource and battery limitations of the mobile devices, training a linear model would be much easier.

3. Methodology

3.1. Dataset

The utilized dataset [2] includes seven physical activities, which are walking, standing, jogging, sitting, biking, walking upstairs, and walking downstairs, performed by ten participants for $3-4$ min. Data were collected from five different positions, which are left pocket, right pocket, belt, upper arm, and wrist. In our daily life, we use these positions changing according to our activities. For example, we use upper arm position specifically when we do jogging or similar activities, and as another example, nowadays we start to use smart watches, so wrist position has become important. To collect data, five smart phones (Samsung Galaxy S2) were placed in these positions with a fixed orientation and sampled at the same time. The orientation of the smart phones was portrait except for the belt position for which it was landscape. Data sampling rate was 50 samples per second. This value was also used in previous studies [1, 12] and sufficient to capture human activities. More details about the dataset can be found in [2].

3.2. Preprocessing

Activity data is segmented into windows before extracting features. In this study, we use a sliding window approach with a 50% overlap for feature extraction, where the window size is 2 s. We selected sliding window of 2 s based on existing works in the literature [12, 18], and 50% overlap has been reported to lead to higher recognition rate results and they are not prone to missing important events [19, 20]. However, in [20], it was also reported that when overlapping windows are used together with k-fold cross validation, the performance of HAR systems is overestimated. We should note that in this paper our aim is to explore the importance of different parameters on the recognition performance and to make comparisons with the previous studies. Since we focus on many parameters, we did not parametrize the window overlap ratio, instead we used the most common value in the related studies for this type of locomotion activities.

3.2.1. Sensors

In the dataset, all the motion sensors available on the smart phones were utilized: accelerometer, gyroscope, magnetometer, and the linear acceleration sensors (this is a virtual sensor and its value is calculated by subtracting gravity from the accelerometer which can be done on phone at real-time or afterwards manually, note that gravity value is a constant and equal to 9.81 m/s^2)[2]. These have been already used in the literature in combination with each-other or alone [7, 12]. To generalize, we also examine all the combinations of these sensors, like accelerometer-gyroscope, gyroscope-magnetometer, accelerometer-gyroscope-magnetometer, etc., except the combinations in which accelerometer and linear acceleration are found at the same time. The main reason to remove them is the production of linear acceleration from the acceleration sensor. Each motion sensor comprises three dimensions which are x-axis, y-axis, and z-axis. In our work, to prevent the orientation changes, we added a fourth dimension, called the magnitude. The effect of orientation changes into activity recognition performance was motivated in [9]. In addition, in [21], the authors proposed a robust AR system by considering essentially orientation variations. Magnitude value is simply the square-root of sums of squares of each axis. Hence, the sensor readings are represented in four dimensions: x-axis, y-axis, z-axis, and the magnitude values. While extracting our feature-sets from raw data, we also calculated pitch and roll values from the accelerometer and linear acceleration sensors. The use of pitch and roll values was proposed in [22] and

[2]Gravity (2017). SensorEvent [online]. Website https://developer.android.com/reference/android/hardware/SensorEvent#values [accessed 10 June 2020].

their formulas are presented in Equation 1, respectively. It is shown [22] that there were only minor differences between calculated pitch, roll values, and orientation sensor. Hence, these are not the exact values, rather they are rough approximations.

$$\beta = (180/\pi)\arctan(y/g, z/g) \quad \text{and} \quad \alpha = (180/\pi)\arctan(x/g, z/g) \tag{1}$$

However, we see that their contribution to the recognition performance is low, and one can choose not to add them to decrease computational power need for especially in online activity recognition.

3.2.2. Feature sets

We selected seven feature-sets as shown in Table 1. These individual features are selected because they are suitable for running on mobile phones and wearables such as a smart watch and they were proposed to be used for context recognition in the survey paper [23]. As identified in the survey paper, most of these features have been used in one of such combinations in the previous studies; however, all of them are not used in one single study.

Table 1: Feature sets.

Featureset	*Features in the set*
FS1	Mean, standard deviation
FS2	Mean, standard deviation, range, min, max
FS3	Mean, median, standard deviation, range, min, max
FS4	Mean, standard deviation, RMS, integration, correlation, absolute difference
FS5	Entropy, spectral energy, sum of first five FFT coefficients
FS6	Median, ZCR, RMS
FS7	Variance, ZCR, RMS

$FS1$, $FS6$, and $FS7$ correspond to $FS1$, $FS2$, and $FS3$ in [2], respectively and $FS5$ is extracted from $FS4$. The definitions of these features and their usage in the literature are described as follows:

- Mean: The average value of all samples is called as mean. It can be calculated with minimal cost (both time and space) [23].
- Median: It is the middle value of sorted data samples [23].
- Standard deviation: This is the square root of the variance. In the literature [23], it is used to recognize movements.
- Variance: It is calculated by taking the average of the squared differences from the mean. Variance gives us an idea of the spread but with standard deviation, we can reach the exact distance from the mean.
- Range: The difference between the maximum and the minimum of data samples is called as the range of these samples. It is useful to separate similar activities, such as running and walking.
- Min: Minimum value in the data sample.
- Max: Maximum value in the data sample.
- Integration: This feature is commonly used for accelerometer sensor to understand speed and distance by measuring the signal under the data curve.

- Correlation: It is calculated by using Equation 2 also known as Pearson's product-moment coefficient. Correlation is useful to separate activities which have translations into a single dimension [23].

$$\rho_{x,y} = \frac{cov(x,y)}{\sigma_x \sigma_y} \tag{2}$$

- Absolute difference: Sum of the differences between each sample of the magnitude and the average of the window divided by the number of data points.
- RMS: Root mean square. The square root of the sum of the data for each window over the size of the data samples (Eq. 3). It is mainly used to recognize gestures [23].

$$x_{RMS} = \sqrt{\frac{x_1^2 + x_2^2 + ... + x_n^2}{n}} \tag{3}$$

- ZCR: Zero crossing rate. The number of points where a signal crosses through a specific value corresponding to the half of the signal range [23]. This specific point is the mean of a window segment [2] for our work. Thus, the number of zero crossing is the number of times this signal crosses that specific point [23].
- Spectral energy: It is the squared sum of spectral coefficients of the signal over the length of the sample window [23].
- Sum of FFT coefficients: Here, we use the first five FFT coefficients because they composed of main frequency component [23].
- Entropy: It is computed using the normalized information entropy of the discrete FFT coefficient magnitudes excluding the DC component. In the literature, this feature is used to distinguish activities which have similar energy levels.

3.3. Classifiers and validation

We use four commonly-utilized classifiers implemented in Scikit-Learn platform based on Python language. These four classifiers, namely Naive Bayes, SVM, decision tree, and random forest have been extensively used in the literature and have been shown to produce good classification performance [1].

As the validation method, we used 10-fold cross-validation. Data are divided into 10 sets; 9 of them were used to train and 1 for testing. It is repeated ten times, each time with a different set for testing. Thus, all sets are used both for training and testing. We used the stratified version of cross-validation which means that each set of data has similar length.

4. Performance analysis

In this section, we present the results of our classification analysis. As mentioned, we have different parameters: 7 feature-sets, 4 classifiers, 11 sensor combinations, and 5 positions. We analyze the performance of all the combinations of parameters. All results are collected on a Samsung S5 ultrabook with i5 – 3317U CPU, 1.70 GHz processor, 4 GB RAM and 64 bit operating system. For implementation, we used Python 2.7 with Scikit-learn library version 0.18.1 [24] and for statistical analysis, we used 17. All the features are normalized using min-max normalization. As the performance metric, F1-score, also called F-measure, is used, which is the harmonic mean of precision and recall.

Optimizing classifiers' parameters is an important issue since it may affect the recognition results. There are two general techniques: grid search and random search. In this part, we use grid search to find optimum parameters for our classifiers. However, this process is time-consuming. Thus, we decided to apply parameter tuning on ACC-G-M sensor combination for FS7 which has the best performance according to our analysis with default parameter values. For decision tree, the minimum number of samples for a split in an internal node is calculated as 2, the minimum numbers of samples to be a leaf is determined as 10, and the best function in order to find best split calculation is determined as log2 of number of features. For SVM, the best kernel type is determined as rbf, the corresponding best penalty parameter is found as 10, and gamma value as 0.2. For random forest, the minimum number of samples required in a leaf node is found as 3, the number of trees in the forest as 20, best measure to determine quality of a split as entropy, and to be able to split minimum number of samples is found as 10. The other parameters which are not specified here are used with their default values. We did not tune GNB's parameter as it has only one, which is priors.

In Table 2, we present a comparison between the results obtained with the tuned parameters and the default parameters. We see that default values of parameters also perform as well as the tuned version. Although decision tree and the random forest results are similar, SVM performs better with the tuned parameters. However, this process is time-consuming and when we compare the results of classifiers with default and tuned parameters, performance difference was small. Hence, we use the default parameters for classifiers in Scikit-learn library version 0.18.1.

Table 2: ACC-G-M with respect to FS7.

Classifier	*Default*	*Tuned*
SVM	0.85	0.90
DT	0.79	0.78
RF	0.87	0.88

This work does not provide an online activity recognition solution. However, when the solution is applied on a smart phone, measuring the run times of recognition steps may be critical due to the limited battery capacity. In another study [25], we measured the run times as well as the resource consumption, both on smart phones and smart watches and showed that both feature extraction and classification can be performed real-time on such devices.

4.1. Impact of feature selection

As mentioned, we selected seven feature-sets as shown in Table 1. However, it is clear that to analyze a dataset with machine learning techniques, we need to select features from data to separate irrelevant features. Reduction of computation time during the training phase and avoidance from overfitting are among the advantages of feature selection. Here, we use three main feature selection techniques. Recursive feature elimination removes recursively features depending on the weights until reaching a specific number of features. As the second method, univariate feature selection with chi^2 is used. The final method is the feature selection with random forest feature importance with the mean decrease accuracy, which is based on permutation of features and towards these results, it measures the impact of each one on the model's accuracy. We calculated top 50 features among 300 (60 features $\times$ 5 sensors) for each feature selection technique.

In Table 3, we present the selection of 15 features per position by all three methods. Next to each feature what is shown is the number of times this feature is selected by the three methods. All-cases column shows the number of times a feature selected in five positions and when all data from different positions are combined.

Standard deviation (STD) is the most commonly selected feature by all, but it ranks as the third one for the right pocket position. Other most commonly selected features are median, range, RMS, variance, energy, mean, max, min. As we will discuss the impact of feature-set in Section 4.3, FS3, FS2, FS7, and FS6 which include these most commonly selected features score as the highest sets in recognizing these set of activities.

Table 3: Mostly selected features, per position and all positions.

	Belt		Left pocket		Right pocket		Arm		Wrist		All positions		All cases	
1	STD	21	STD	19	Mean	19	STD	19	STD	20	STD	21	STD	121
2	Range	17	Energy	19	Median	19	Max	17	Range	18	Median	19	Range	97
3	Energy	15	Range	16	STD	16	RMS	15	RMS	16	Range	16	Energy	97
4	Variance	15	RMS	15	Range	15	Energy	15	Max	14	Energy	16	RMS	83
5	Min	13	Median	14	Min	14	Range	14	Variance	14	Variance	14	Variance	83
6	Max	13	Min	14	Max	14	Variance	14	Energy	13	ABS	13	Median	82
7	RMS	13	Variance	13	Integration	13	Median	13	Median	12	RMS	12	Min	69
8	Median	12	Integration	8	Correlation	8	Min	11	Min	10	Min	10	Max	65
9	Mean	7	ABS	8	ABS	8	Mean	8	Mean	9	Mean	9	Mean	53
10	ABS	7	Mean	6	RMS	6	ABS	7	ABS	8	Max	8	ABS	51
11	Entropy	7	Max	6	ZCR	6	Integration	6	Integration	4	Entropy	5	Integration	33
12	Integration	5	coefSum	5	Energy	5	Correlation	5	Correlation	3	Integration	2	Entropy	19
13	Correlation	2	Entropy	3	coefSum	3	coefSum	2	coefSum	2	Correlation	2	Correlation	17
14	coefSum	2	Correlation	2	Entropy	2	ZCR	0	Entropy	2	coefSum	2	coefSum	15
15	ZCR	0	ZCR	0	Variance	0	Entropy	0	ZCR	0	ZCR	0	ZCR	0

4.2. Impact of position information

In this section, we discuss the classification performance for each position individually: belt, left-pocket, right-pocket, upper arm, and wrist. In order to understand the performance of feature-sets according to different positions, we evaluated them using four classifiers and all sensor combinations. In this section, we present only the results with the accelerometer sensor due to page limitations, for ease of presentation and to focus on the effect of feature-set. We should note that, achieved results with other sensors were lower and they can be accessed in [26]. In Tables 4a–4d, we see corresponding results in terms of f-measure, with GNB, SVM, RF, and DT classifiers, respectively.

If we analyze the results one by one per classifier, in Table 4a, the best feature-sets are $FS6$ and $FS7$ for the belt and pocket positions, while in the upper arm and wrist positions they exhibit very similar performance with the $FS3$ and $FS4$. All feature-sets perform better in pocket positions compared to other positions, between 80% and 90%. Pocket position can capture the leg movements and for this set of activities, it has been the most commonly used position in the literature.

Arm and wrist positions follow the pocket position with 80% performance. The order is as $FS2$, $FS3$, $FS1$, $FS4$, $FS6$, $FS7$, $FS5$ for the upper arm position, while it is as $FS4$, $FS6$, $FS7$, $FS3$, $FS1$, $FS2$, $FS5$ for the wrist position, in terms recognition performance. The lowest F-measure results are observed in the belt position, close to 70%. Compared to the other positions, the phone is more stationary. When we compare the performance of the feature-sets, $FS6$ and $FS7$ are the best performing sets, achieving a score of 71% and 73%, respectively. When we analyze the confusion matrices (they are not presented here due to space limitations) in the belt position and in the right pocket positions, walking downstairs activity is confused with walking and walking upstairs activities. However, in the belt position, the walking activity is also confused with downstairs and upstairs activities, while this is not true for the pocket position and the confusion rate of the

walking activity is less. As expected, standing and sitting activities are confused with each other in the belt position, while this is not the case for the pocket position.

When we look at the performance of SVM, Table 4b, we observe that results are better than the performance of the GNB classifier. $FS6$ and $FS7$ again perform well, but are not always the best (except left pocket). $FS3$ and $FS4$ are better performing in the wrist and upper arm positions. At upper arm, feature-sets are ordered as $FS2$, $FS3$, $FS1$, $FS4$, $FS6$, $FS7$, $FS5$ and at wrist, feature-sets ordered as $FS4$, $FS3$, $FS1$, $FS2$, $FS6$, $FS7$, $FS5$. In belt and left pocket positions, the performance of $FS3$ and $FS4$ is similar to that of $FS6$ and $FS7$. The only difference between $FS6$ and $FS7$ is that one includes median while the other has variance. $FS6$ and $FS7$ include only three features, while $FS3$ and $FS4$ have six and seven features, respectively. $FS1$ and $FS2$ also exhibit similar performance values, differing up to 5% lower rate, and these two sets also include very simple features to compute. $FS5$, on the other hand, exhibits lower performance in all positions. $FS5$ includes mostly the frequency domain features: entropy, spectral energy, the sum of first five FFT coefficients. These features require more computation compared to the other simple time-domain features; hence, it is not required to use them for this set of activities, particularly in a real-time recognition application running on phones.

In Table 4c, we see the performance results with the random forest classifier. The results are similar to SVM and better than GNB. However, SVM achieves higher performance for the belt position. In the random forest case, $FS3$ is the best performing set in all positions. However, the difference is only between 0% and 3% compared to $FS2$ and $FS7$ for all positions. Finally, in Table 4d, when we look at the results of the decision tree classifier, performance is lower than random forest and SVM. There is no feature-set that outperforms the others in all positions. For example, $FS7$ performs the best in right pocket position, while $FS3$ is the best in the upper arm. However, a similar trend is visible: $FS5$ has worse performance than the other feature-sets, except the pocket positions. $FS5$ exhibits worse performance also in pocket positions with SVM.

Table 4: Impact of using different feature-sets with four classifiers

	FS1	FS2	FS3	FS4	FS5	FS6	FS7
Belt	0,66	0,67	0,68	0,68	0,64	0,71	0,73
Lpocket	0,83	0,84	0,85	0,83	0,84	0,87	0,87
Rpocket	0,85	0,85	0,87	0,85	0,85	0,89	0,87
Uarm	0,79	0,8	0,8	0,78	0,74	0,78	0,77
Wrist	0,77	0,77	0,79	0,8	0,74	0,8	0,79

(a) Impact of feature-sets with GNB classifier

	FS1	FS2	FS3	FS4	FS5	FS6	FS7
Belt	0,69	0,74	0,76	0,75	0,65	0,75	0,75
Lpocket	0,86	0,87	0,89	0,86	0,83	0,9	0,9
Rpocket	0,87	0,88	0,9	0,9	0,84	0,9	0,91
Uarm	0,8	0,82	0,82	0,79	0,74	0,79	0,79
Wrist	0,82	0,82	0,83	0,85	0,75	0,81	0,81

(b) Impact of feature-sets with SVM classifier

	FS1	FS2	FS3	FS4	FS5	FS6	FS7
Belt	0,71	0,72	0,74	0,73	0,71	0,74	0,7
Lpocket	0,87	0,9	0,9	0,87	0,86	0,89	0,89
Rpocket	0,88	0,89	0,9	0,88	0,85	0,87	0,9
Uarm	0,77	0,82	0,82	0,76	0,71	0,76	0,82
Wrist	0,8	0,82	0,85	0,8	0,77	0,8	0,82

(c) Impact of feature-sets with RF classifier

	FS1	FS2	FS3	FS4	FS5	FS6	FS7
Belt	0,72	0,71	0,72	0,7	0,65	0,67	0,68
Lpocket	0,84	0,83	0,84	0,82	0,83	0,8	0,84
Rpocket	0,82	0,84	0,83	0,82	0,8	0,84	0,89
Uarm	0,73	0,77	0,79	0,75	0,69	0,72	0,79
Wrist	0,8	0,78	0,77	0,78	0,72	0,77	0,8

(d) Impact of feature-sets with DT classifier

4.3. Analysis with all positions

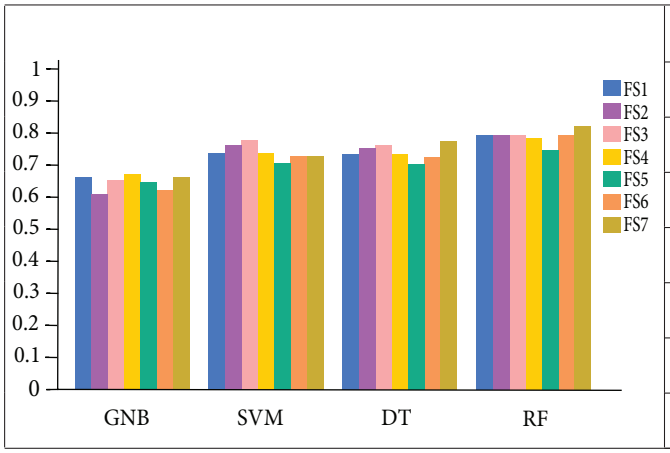
Having investigated the performance by positions, in this section we explore the position-independent recognition performance results. In Section 4.2, the models were built and validated per position data, while in this section we combine all the data from five positions and build the models. If position information is known, the recognition performance may increase for some positions, such as a pocket; however, in general, average performance does not differ too much, as shown in the literature [22]. Moreover, it may not be possible to have the position information; hence, a general classifier can be more practical in a real-time recognition application. In Section 4.3.1, we examine the effect of feature-sets, and in Section 4.3.2, we examine the effect of sensor/sensors on the recognition performance.

4.3.1. Impact of feature set

In this section, we present the results of the accelerometer sensor, with all the feature-sets, and with the four classifiers, we investigate the effect of feature-sets. The results for other sensors can be found in [26] .

As shown in the figure in Table 5, random forest classifier performs as the best for each feature-set. It achieves the maximum measure with $FS7$: 82%. SVM achieves 77% performance with $FS3$, and decision tree achieves 76% with $FS4$. Except for GNB, the other classifiers' performances are higher than 70% with all the feature-sets. In Table 5, we show the feature ranking with all the classifiers. When a random forest is used, $FS1$, $FS2$, $FS3$, and $FS6$ achieve the same rates, while $FS7$ achieves 3% higher measure. With SVM, the ranking is as $FS3$ (77%), $FS2$ (76%), $FS1$, and $FS4$ with 73%, and so on. Different from these, $FS4$ achieves the highest measure with the decision tree classifier, 76%. If we compare our findings with the results of Section 4.2, $FS3$ is again the best performing feature-set among all the classifiers and $FS5$ are the worst ones. Here the ranking is as: $FS3$, $FS1$, $FS4$, $FS2$ and $FS7$, $FS6$, $FS5$, considering the performance of all the classifiers, while it was $FS3$, $FS7$, $FS2$, $FS6$, $FS4$, $FS1$, $FS5$, per position results. These results are only for the accelerometer sensor. However, we present the rankings considering the other sensor and/or sensor combinations in Section 4.3.2.

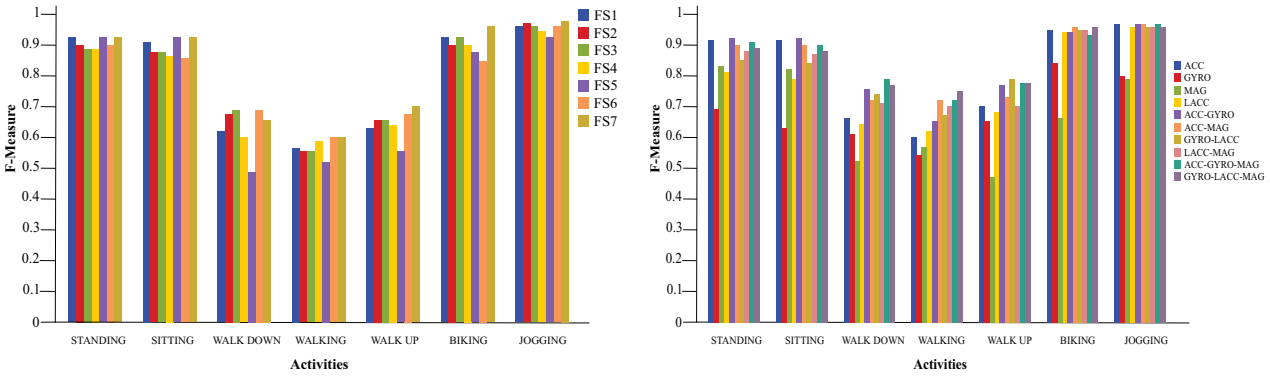
Table 5: Feature ranking with all positions, per classifier, using accelerometer.



	Feature Sets						
Classifiers	FS1	FS2	FS3	FS4	FS5	FS6	FS7
GNB	2	7	4	1	5	6	2
SVM	3	2	1	3	7	5	5
DT	3	3	2	1	3	7	6
RF	2	2	2	6	7	2	1
Sum of Ranks	10	14	9	11	22	21	14
Final Ranking	2	5	1	3	7	6	5

To show the results per activity, we present the results of the Random Forest using the accelerometer in Figure 1a. For other sensor combinations and classifiers, results are presented in [26]. As we discussed earlier, mostly confused activities are the walking upstairs, downstairs, and the walking activity and their scores are

lower. However, standing and sitting activities are classified with more than 90% performance and jogging and biking activities are classified with more than 95%.



(a) Impact of feature-sets with random forest and accelerometer on all activities

(b) Impact of sensors with random forest classifier with $FS7$

Figure 1. Impact on activities

4.3.2. Impact of sensor and fusion

To investigate the effect of sensors and their combinations, we present the results with the best-performed classifier that is found in Section 4.3.1 which is the Random Forest. However, performance with all the classifiers can be accessed in [26].

To show the impact of sensors on classifying individual activities, we present the results of Random Forest using $FS7$ in Figure 1b. For other feature-sets and classifiers, results are presented in [26]. When sensors are combined, performance recognition of walking activities also increases up to 80%, other activities are classified with more than 90% performance.

Results are presented in Table 6 in terms of f-measure for all feature-sets. If we compare the performance when a single sensor is used, then it is clear that accelerometer alone performs the best with all feature-sets. $FS3$ achieves 77% performance while $FS2$ is 76%, and $FS6$ and $FS7$ present 72%. This is followed by the linear acceleration sensor. The maximum recognition performance is achieved with $FS6$: 69%. Gyroscope alone performs close to linear acceleration, 64% with $FS6$. On the other hand, magnetometer achieves the lowest measure: maximum 58% with $FS7$ and 44% with $FS2$ and $FS3$.

Table 6: Impact of sensors and their fusion.

	ACC	G	LA	M	ACC-G	ACC-M	G-M	G-LA	LA-M	ACC-G-M	G-LA-M
FS1	0.79	0.67	0.76	0.56	0.85	0.82	0.77	0.84	0.81	0.87	0.86
FS2	0.79	0.68	0.76	0.57	0.85	0.82	0.77	0.83	0.81	0.86	0.85
FS3	0.79	0.69	0.77	0.57	0.86	0.84	0.78	0.84	0.82	0.86	0.85
FS4	0.78	0.68	0.76	0.56	0.85	0.82	0.78	0.84	0.8	0.86	0.85
FS5	0.74	0.61	0.71	0.54	0.8	0.77	0.73	0.8	0.76	0.81	0.81
FS6	0.79	0.72	0.79	0.6	0.86	0.81	0.8	0.87	0.83	0.87	0.88
FS7	0.82	0.68	0.78	0.67	0.85	0.85	0.79	0.83	0.83	0.86	0.86

When we consider the double combinations of the sensors, accelerometer-gyroscope combination achieves the best performance with all the feature-sets. Moreover, $FS1$ with this combination achieves the best performance with all the combinations. This is followed by linear acceleration and gyroscope combination. Compared to the single use of sensors, in double combinations, performance increases. For example, accelerometer alone achieves 77% performance, while the accelerometer-gyroscope combination achieves 85%. When we look at the triple combinations of sensors, they perform slightly better than the double combinations. However, the difference is only 1–2%. The highest performance is 88%. The combination of acceleration, gyroscope, and magnetometer sensors perform very similarly to the combination of linear acceleration, gyroscope, and magnetometer sensors.

In Table 7, we present the sensor rankings. Each row shows the ranking of sensors according to their performance when a specific feature-set is used. This was also shown in Table 6. Accelerometer, gyroscope, and magnetometer combinations perform the best with all feature-sets. This is followed by linear acceleration, gyroscope and magnetometer combinations. The combination of accelerometer and gyroscope follows that, which is then followed by the linear acceleration and gyroscope. It is clear that accelerometer and linear acceleration sensors are the dominating sensors. However, as shown in the literature, linear acceleration sensor consumes more power than the accelerometer. Gyroscope and magnetometer support the performance. However, the effect of the gyroscope is definitely higher. If battery/power consumption is a concern, then a single sensor should be used and this should be the accelerometer. However, supporting sensors, such as the gyroscope, can be turned on/off when needed.

Table 7: Ranking of sensors and sensor combinations with random forest.

	ACC	GYRO	MAG	LACC	ACC & GYRO	ACC & MAG	GYRO & MAG	GYRO & LACC	LACC & MAG	ACC & GYRO & MAG	GYRO & LACC & MAG
FS1	7	10	11	9	3	5	8	4	6	1	2
FS2	7	10	11	9	2	5	8	4	6	1	2
FS3	7	10	11	9	1	4	8	4	6	1	3
FS4	7	10	11	9	2	5	7	4	6	1	2
FS5	7	10	11	9	3	5	8	3	6	1	1
FS6	8	10	11	8	4	6	7	2	5	2	1
FS7	7	10	11	9	3	3	8	5	5	1	1
Sum of Ranks	50	70	77	62	18	33	54	26	40	8	12
Final ranking	7	10	11	9	3	5	8	4	6	1	2

Additionally, we present the feature rankings in Table 8. Each row shows the ranking of features according to their performance when a specific sensor/sensors are used. When we consider all sensors and sensor combinations, $FS7$ and $FS3$ are among the best performing feature-sets. However, $FS6$ ranks the best. Again, $FS5$ is the worst performing feature-sets.

4.4. Impact of all parameters

Up to this section, we investigate the performance of parameters by comparing one to another. Even if we can have an idea about the effects of each parameter, we cannot see the global picture because of the dependency of too many parameters. In this section, using statistical analysis, we examine the effects of each parameter (sensors, positions, feature-sets, classifiers, activities) on the performance and we give exact impact values. After

Table 8: Feature ranking with different sensors using random forest.

	FS1	FS2	FS3	FS4	FS5	FS6	FS7
ACC	2	2	2	6	7	2	1
GYRO	6	3	2	3	7	1	3
MAG	5	3	3	5	7	2	1
LACC	4	4	3	4	7	1	2
ACC-GYRO	3	3	1	3	7	1	3
ACC-MAG	3	3	2	3	7	6	1
GYRO-MAG	5	5	3	3	7	1	2
GYRO-LACC	2	5	2	2	7	1	5
LACC-MAG	4	4	3	6	7	1	1
ACC-GYRO-MAG	1	3	3	3	7	1	3
GYRO-LACC-MAG	2	4	4	4	7	1	2
Sum of ranks	37	39	28	42	77	18	24
Final ranking	4	5	3	6	7	1	2

collecting all F-measure results according to all possible cases, we constructed a new data file which contains all of the parameter combinations and F-measures achieved with that combination. For impact analysis, we use Minitab statistical software.

Firstly, we convert F-measure values into categorical types for the analysis. For this purpose, we suppose that F-values within the range of 0–50% are considered bad, 50–60% are considered low, 60–70% medium, 70–80% good, 80–90% as very good, and finally 90–100% are considered excellent. To examine the effect of sensors, features, positions, classifiers on the target (F-measure), as the target value is categorical and ordinary, we used ordinal logistic regression. We choose 0.05 for significance level (α) for all analyses presented in this section. It is known that a parameter is statistically significant if the corresponding P-value is lower than 0.05. We observe the following values for each parameter: sensors (0.952), positions (0.000), features (0.000), and classifiers (0.147). We see that only position and feature are significant. Moreover, we repeated this analysis by not converting F-values into categorical types. Since target value is continuous, we use ANOVA analysis. As a result of this analysis, we see that all parameters are significant for the target value. When we look in detail (Table 9), we see that, the ranking is as follows: sensor, position, classifier, feature. According to these results, we can say that when we convert target values into categorical type, we loose details; thus, we observed that sensor and classifier are not significant, which is not true as we see in Table 9.

We also investigate the parameter combinations and the exact effect ratio on the target. For this purpose, we use adjusted R squared and predicted R squared statistical terms. Adjusted squared R term defines the variance percentage on the target value, which is explained by the input variables. In other terms, it defines variance percentage in response which can be explained by the model. On the other hand, predicted R squared explains how well a model predicts new observations. For our analysis with activity-independent ANOVA analysis, the order given by the adjusted R squared and predicted R squared terms are the same, which means that the given impact order by the two terms are coherent. Table 9 should be interpreted as follows: by knowing just the feature, the model could predict 2.94% of the target value and by knowing the sensor-position-feature-classifier at the same time, the model could correctly predict the 98.67% of the target.

Table 9: Activity-independent ANOVA analysis.

Parameters	$R-sq(adj)$	$R-sq(pred)$
sensor-position-feature-classifier	98.67%	98.28%
sensor-position-classifier	90.27%	88.64%
sensor-position-feature	81.81%	79.73%
sensor-position	79.61%	75.84%
sensor-classifier	63.80%	62.71%
sensor-feature-classifier	62.46%	58.05%
sensor-feature	57.23%	54.95%
sensor	54.78%	54.42%
position-classifier	23.58%	22.53%
position-feature-classifier	21.65%	20.75%
position-feature	17.87%	16.91%
position	15.90%	15.57%
feature-classifier	10.12%	8.40%
classifier	7.85%	7.55%
feature	2.94%	2.44%

4.5. Discussion

In this section, we summarize our findings and discuss possible improvements.

- $FS3$, $FS7$, $FS6$, and $FS2$ perform the best while $FS5$ is the worst: When we consider the performance with different feature-sets, it is observed that $FS3$, $FS7$, $FS6$, and $FS2$ perform the best. $FS3$ includes the mean, median, standard deviation, range, min, max. The only difference between $FS3$ and $FS2$ is the addition of the median value as a feature, but this definitely increases the classification performance. $FS7$ includes variance, ZCR, RMS as the features which are totally different from $FS1$ and it includes variance while $FS6$ includes the median beside the other two features. $FS5$ (entropy, spectral energy, the sum of first five FFT coefficients) is the worst performing sets. Hence, these results show that with easy-to-compute features, high recognition rates can be achieved. Additionally, when we consider real-time recognition on phones, these simple features can contribute to resource savings, such as the battery, as we discussed in a recent study [25].
- Accelerometer is the dominating sensor: When we discuss the impact of sensors on the recognition performance, accelerometer is the best performing sensor. In some cases, linear acceleration also performs well. However, gyroscope and magnetometer do not perform as well as them when used alone.
- Trade-off with using more sensors and battery consumption: As we discussed fusing information from different sensors increases the recognition performance. On the other hand, this also increases the computational burden. Since more sensors will be sampled, there will be a burden on the battery consumption. Hence, more sensors should be used when necessary. It would be interesting to explore the dynamic sampling of sensors when needed. In a recent paper [25], we investigate the resource consumption of different classifiers in terms of battery, memory and CPU cycles.
- Different phone positions: When we consider different phone positions, highest scores are achieved in the pocket positions, close to 90% recognition performance. Arm and wrist positions follow the pocket

position with 80% performance results. The lowest results are observed in the belt position, close to 70%. Compared to the other positions, the phone is more stationary and its orientation was different. In this position, it was difficult to differentiate walking activity from walking downstairs and upstairs activities, while this is not true for the pocket position and the confusion rate of the walking activity is less. Additionally, standing and sitting activities are confused with each other in the belt position, while this is not the case for the pocket positions. However, we believe that it is necessary to test the performance with different positions since people carry phones in very different positions.

- Recognition performance of individual activities: Most commonly confused activities are walking upstairs, downstairs, and the walking activity and their scores are lower. These three confused activities have very similar patterns and it may be necessary to use other sensors such as pressure sensor to differentiate the elevation. However, jogging and biking activities are classified with more than 95% accuracy.
- ANOVA results reveal that sensor has the highest impact on the performance, which is followed by a position, classifier, and feature-set. Hence, these two parameters can be explored with different values when studying human activity recognition with mobile devices.

5. Conclusions and future work

In this paper, we explored the parameter space of human activity recognition using motion sensors available on smart phones. We use a dataset collected from ten participants at five different positions, with four different sensors, including seven different motion activities. First, we analyze the performance of activity recognition according to the five different positions of the phone. Looking at the performance of various feature-sets, we show that feature-set $FS3$, which includes mean, median, standard deviation, range, min, max, gives the best performance. Since position information may not be always available, or phones can be carried in different positions, we investigate the performance with no position information, which means that data from all positions are combined and analyzed. Accelerometer is found to be the best-performing sensor and in terms of feature-sets, $FS3$ is again the best performing feature-set among all the classifiers. In the final part of the paper, we make an ANOVA analysis, to investigate the impact of different parameters on the recognition scores. The results show that sensor has the highest impact on the performance, which is followed by position, classifier, and feature-set. As a future work, we plan to extend our analysis to other datasets, particularly those including multiple positions and sensor information. We will also extend the activity dataset to include more complex activities and also the device set to include smart watches. Furthermore, we want to apply new classification methods based on ensemble learning techniques and deep learning techniques not yet applied in this domain.

Acknowledgment

This work is supported by Galatasaray University Research Fund under Grant Number 17.401.004.

References

[1] Shoaib M, Bosch S, Incel OD, Scholten H, Havinga P. A survey of online activity recognition using mobile phones. Sensors 2015; 15 (1): 2059-2085. doi: 10.3390/s150102059

[2] Shoaib M, Bosch S, Incel OD, Scholten H, Havinga P. Fusion of smartphone motion sensors for physical activity recognition. Sensors 2014; 14 (6): 10146-10176. doi: 10.3390/s140610146

[3] Plötz T, Hammerla NY, Olivier PL. Feature learning for activity recognition in ubiquitous computing. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence; Barcelona, Catalonia, Spain; 2011. pp. 1729-1734.

[4] Niazi AHK. A study in human activity recognition. PhD, University of Georgia. Athens, GA, USA, 2016.

[5] Bulling A, Blanke U, Schiele B. A tutorial on human activity recognition using body-worn inertial sensors. ACM Computing Surveys (CSUR) 2014; 46 (3): 33. doi: 10.1145/2499621

[6] Gjoreski H, Lustrek M, Gams M. Accelerometer placement for posture recognition and fall detection. In: Intelligent Environments (IE), 7th International Conference on Intelligent Environments; Nottingham, United Kingdom; 2011. pp. 47-54.

[7] Altun K, Barshan B. Human activity recognition using inertial/magnetic sensor units. In: International Workshop on Human Behavior Understanding; Berlin, Heidelberg, Germany; 2010. pp. 38-51.

[8] Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: International Workshop on Ambient Assisted Living; Berlin, Heidelberg, Germany; 2012. pp. 216-223.

[9] Su L, Zhang D, Li B, Guo B, Li S. Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In: International Conference on Ubiquitous Intelligence and Computing; Berlin, Heidelberg, Germany; 2010. pp. 548-562.

[10] Susi M, Renaudin V, Lachapelle G. Motion mode recognition and step detection algorithms for mobile phone users. Sensors 2013; 13 (3): 1539-1562. doi: 10.3390/s130201539

[11] Anjum A, Ilyas MU. Activity recognition using smartphone sensors. In: IEEE 10th Consumer Communications and Networking Conference (CCNC); Las Vegas, NV, USA; 2013. pp. 914-919.

[12] Wu W, Dasgupta S, Ramirez E et al. Classification accuracies of physical activities using smartphone motion sensors. Journal of Medical Internet Research 2012; 14 (5): e130. doi: 10.2196/jmir.2208

[13] Martín H, Bernardos Ana M, Iglesias J, Casar José R. Activity logging using lightweight classification techniques in mobile devices. Personal and Ubiquitous Computing 2013; 17 (4): 675-695. doi: 10.1007/s00779-012-0515-4

[14] Zhang L, Suganthan P. Benchmarking ensemble classifiers with novel co-trained kernal ridge regression and random vector functional link ensembles [Research Frontier]. IEEE Computational Intelligence Magazine 2017; 12 (4): 61-72. doi: 10.1109/MCI.2017.2742867

[15] Liu L, Wang S, Su G, Huang ZG, Liu M. Towards complex activity recognition using a Bayesian network-based probabilistic generative framework. Pattern Recognition 2017; 68: 295-309. doi: 10.1016/j.patcog.2017.02.028

[16] Oniga S, Suto J. Human activity recognition using neural networks. In: Proceedings of the 2014 15th International Carpathian Control Conference (ICCC); Velke Karlovice, Czech Republic; 2014. pp. 403-406.

[17] Ordóñez FJ, Roggen D. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors 2016; 16 (1): 115. doi: 10.3390/s16010115

[18] Preece SJ, Goulermas JY, Kenney LPJ, Howard D. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. IEEE Transactions on Biomedical Engineering; 56 (3): 871-879. doi: 10.1109/TBME.2008.2006190

[19] Shoaib M. Sitting is the new smoking: online complex human activity recognition with smartphones and wearables. PhD, University of Twente. Enschede, Netherlands, 2017.

[20] Dehghani A, Sarbishei O, Glatard T, Shihab E. A quantitative comparison of overlapping and non-overlapping sliding windows for human activity recognition using inertial sensors. Sensors 2019; 19 (22): 5026. doi: 10.3390/s19225026

[21] Chen Z, Zhu Q, Yeng CS, Zhang L. Robust human activity recognition using smartphone sensors via CT-PCA and online SVM. IEEE Transactions on Industrial Informatics 2017; 13 (6): 3070-3080. doi: 10.1109/TII.2017.2712746

[22] Coskun D, Incel OD, Ozgovde A. Phone position/placement detection using accelerometer: Impact on activity recognition. In: Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference; Singapore; 2015. pp. 1-6.

[23] Figo D, Diniz PC, Ferreira DR, Cardoso JMP. Preprocessing techniques for context recognition from accelerometer data. Personal and Ubiquitous Computing 2010; 14 (7): 645-662. doi: 10.1007/s00779-010-0293-9

[24] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011; 12: 2825-2830.

[25] Shoaib M, Incel OD, Scholten H, Havinga P. Resource consumption analysis of online activity recognition on mobile phones and smartwatches. In: 36th IEEE International Performance Computing and Communication Conference; New York, USA; 2017. pp.1-20.

[26] Saylam B. A detailed analysis of human activity recognition using smartphone motion sensors. MSc, Galatasaray University, İstanbul, Turkey, 2017.