9th International Conference on Computer and Knowledge Engineering (ICCKE 2019),
October 24-25 2019, Ferdowsi University of Mashhad

Tennis stroke detection using inertial data of a smartwatch

Sara Taghavi, Fardjad Davari, Hadi Tabatabaee Malazi, Ahmad Ali Abin
Faculty of Computer Science and Engineering, G.C.
Shahid Beheshti University, Tehran, Iran
Email: {sa.taghavi, f.davari}@mail.sbu.ac.ir, {h_tabatabaee, a_abin}@sbu.ac.ir

***Abstract*—To assist individuals in sports activities is one of the emerging areas of wearable applications. Among various kinds of sports, detecting tennis strokes faces unique challenges. In this sport the speed of strokes is high, enforcing wearable sensors to have high sampling rates, high-speed bus (to transfer the data to the processor), and the most importantly adequate size of high-speed memory. The constraints encourage researchers to design a custom made hardware to cope with the challenges. The research question that we are trying to address is to show how accurate a commercial smartwatch can detect tennis strokes by using various techniques in machine learning. In this paper, we propose an approach to detect three tennis strokes by utilizing a smartwatch. In our method, the smartwatch is part of a wireless network in which inertial data file is transferred to a laptop where data prepossessing and classification is performed. The data file contains acceleration and angular velocity data of the 3D accelerometer and gyroscope. We also enhanced our method with data prepossessing techniques to elevate data quality. The evaluation of our devised method shows promising results compared to a similar method.**

***Index Terms*—Activity recognition, Sport activity, Tennis strokes, Wearable sensors, Smartwatch.**

I. Introduction

In recent years, analysing sports activities has experienced a change from time-consuming image-based methods to ones based on inertial sensors data for example in cricket to validate the elbow extension in bowling action [1] or in rock climbing for fall detection [2]. Usually, athletes like to use a wearable as *Inertial Measurement Units* (IMUs) for its lightweight and non-intrusive nature during their sports activities.

A reliable activity detection system [3]–[5] can be built by using wearable sensors and machine learning techniques collaboratively. These systems are used in sports such as volleyball, swimming, and rock climbing to prevent injuries, monitor performance, and improve practice routines. For example, they can assist the coach in volleyball to see if a player has put excessive pressure into his shoulders by spiking and can lead him to activities that have less impact on shoulders to prevent injuries [6]. Also identifying and counting falls in rock climbing is useful either as a performance indicator or as a safety measure [2]. Estimating parameters as stroke count per arm, speed per lap, and totally swam distance [7] helps an athlete and a coach to improve practice routines for better efficiency.

From 2014, the *International Tennis Federation* (ITF) was one of the first sports federations who accepted the use of wearable even in competitions [8], [9]. Several researches have been conducted in *Tennis Activity Detection* (TAD) using custom-built wearable. They have been designed in laboratories with embedded MEMS-based inertial sensors and micro-controllers. A miniature wearable device and system are developed and designed for movement and biomet data acquisition and stroke detection [8]. *Büthe* et al. in [10] designed three wireless IMUs to detect tennis strokes and footwork. The first unit is embedded in tennis racket and the other two are placed on player's shoes. An embedded 6-axis sensor product is identified in [11]. The hardware architecture includes three modules of sensor, control, and transmission. Due to a very few publicly available tennis strokes datasets, most of the studies have acquired their own datasets and utilize the custom-built wearable. Unfortunately, their datasets are not published publicly either. The University of Texas at Dallas multi-modal dataset [12] is one of the few publicly available human action recognition datasets that include two tennis activities - serve and forehand swing. This multi-modal dataset for HAR utilizes a depth camera and a custom-built wearable inertial unit.

Among various available wearables, we selected a smartwatch for TAD because it is one of the most popular ones. The selected smartwatch has five embedded sensors including accelerometer, barometer, gyroscope, heart rate and orientation. We utilized accelerometer and gyroscope to detect tennis strokes by applying machine learning techniques. To detect tennis strokes, we faced several challenges. The first one is the fluctuating sampling rate of sensor data [13]. Since the processor is not dedicated to reading sensor data and has the burden of processing other tasks, it simply misses some reading in high sampling rates. The second challenge is the limitation of data transmission rate that leads to slowing down the online data transfer. Thirdly, smartwatches are highly resourced constraint devices regarding memory space to store latest sensor readings. Finally, the processing power adds more constraints on any application.

In this paper, we decided to provide a public domain tennis stroke dataset utilizing an off the shelf smartwatch. Further, we proposed an approach for stroke detection and classification based on the provided dataset. The proposed approach utilizes datastream alignment and signal filtering methods for data pre-processing to overcome poor smartwatch data quality. The well-known *Principal Component Analysis* (PCA) dimension

978-1-7281-5075-8/19/$31.00 ©2019 IEEE

reduction method is also used to reduce dimentionality of data and consequently improve the accuracy of classification. We also utilized importance of features to determine which features are more effective in the classification process. Then, we evaluated the approach on smartwatch dataset and *UTD-MHAD* [12] in three different scenarios and compared the results. The outcomes show that the proposed approach has improved the overall classification accuracy by more than 30%.

The rest of this paper is organized as follows: In the next section, we review some of the well-known related works in tennis activity detection. Our devised smartwatch based tennis activity recognition method is described in Section III. Section IV is dedicated to the performance analysis of our approach. Finally, Section V concludes the work.

II. Related works

There is ongoing research in detecting tennis activities from commercial and academic sectors.

Babolat, a well-known tennis equipment company, developed a system, called *Babolat PLAY* that is embedded in a racket handle. It uses a Bluetooth connection to connect with a smart mobile device. Babolat also has a wrist-worn device, called *Babolat POP* with similar performances as the *Babolat PLAY* [14], [15]. Similar systems attachable to the handle of the racket are manufactured by *Sony* and *Zepp* as well [14].

Tennis activity recognition task can also benefit from image/video processing techniques [16], [17]. *Play Sight* [18] is a tennis analytic system that employs six high definition cameras to draw out information about the game [19].

Our main focus of research in this paper is to use inertial data, therefore in the following, we briefly review some of the outstanding related works.

M. Kos et al. in [8] introduced a system for movement and biomet information such as skin temperature and heart rate data acquisition. They have designed a lightweight wrist-worn device that communicates with the personal computer through a USB connection. On the computer side, a software component is used to download the data from the device and upload it to cloud service. They also provided a *Tennis Stroke Database* (TSD) for their TAD purposes. The TSD includes different players with different levels of proficiency, different environments (indoors, outdoors), court surfaces (clay, hard), different tennis balls, and rackets. The TSD contains 446 strokes; the recordings are a mixture of individual and competitive data sequences. Unfortunately, it is not published as a public dataset. For stroke detection, they had a threshold-based approach to detect the strokes in data streams. The thresholds were defined by observing the accelerometer and gyroscope data. The stroke classification is a threshold-conditional-base method. The algorithm calculates a two-point derivative of the acceleration curves and compares them with a predefined threshold. Then, classifies the strokes according to predefined conditions on gyroscope 3-D sensors axis.

Pei et al. in [11] developed a product that consists of three modules including sensor, controller, and transmission. The product is embedded in tennis racket handle without attachment to the body. The motion information is sent to the mobile phone in real time. For stroke recognition, they took three steps. First, the threshold-based shot detection method that uses fluctuation of acceleration based on moving windows. Second, detected shots from step one are divided into three stroke types including forehand, backhand and serve according to acceleration and angular velocity values. At last, based on the angular velocity information about ball rotation, forehand or backhand are divided into topspin or backspin.

Dhnesh et al. in [19] introduced a platform of wireless measurement sensors, which work in conjunction with software analysis modules to monitor and assess the effectiveness of drills involved in the improvement of tennis serves. The custom-built sensors can measure kinetic parameters in real-time and are attached to the racket and to the player's body. The software module enables visualization of sensor data and evaluates performance.

Although commercial products from various companies such as *Babolat*, *Zepp* and *Sony* are available for the players, the products are usually embedded to a specific racket. Therefore, the players may need to have a wearable which is not bounded to a specific racket. On the other hand, the custom-build wearable such as IMUs are not practical and publicly available for everyone. From these points of view, choosing a wearable such as smart-watch in much convenient for tennis activity detection.

III. Smartwatch data acquisition and stroke detection system

While all the previously reviewed works utilized custom-built IMUs, we used traditional smartwatch sensors. Our devised system is comprised of two parts. The first one is the data collector application which contains different modules to read the sensors data out, save them in a file on the smartwatch, shrink the file into small pieces, and transfer them to laptop through the company's mobile app. The second part is responsible for data pre-processing and activity recognition.

From network point of view, an overview of the system components is depicted in Fig.1. Smartwatch, mobile, and laptop are all in a wireless local network. Smartwatch application communicates with the companion through a message socket protocol and the companion communicates with the laptop (or any other device that can process data, such as cloud service) by a web socket protocol. In the following, these parts are elaborated in more detail.

A. Data acquisition and data file transfer

The *data collector application* (DCA) is designed with five modules including haptic feedback, sensors helper, messaging helper, logger, and client SDK. Fig.2 presents the block diagram of the application.

- Haptic feedback: This module controls the vibrations. We used vibrations to announce the subject who wears the watch, for critical moments like starting and stopping the activity.

Fig. 1: An overview of smart-watch data acquisition and stroke detection

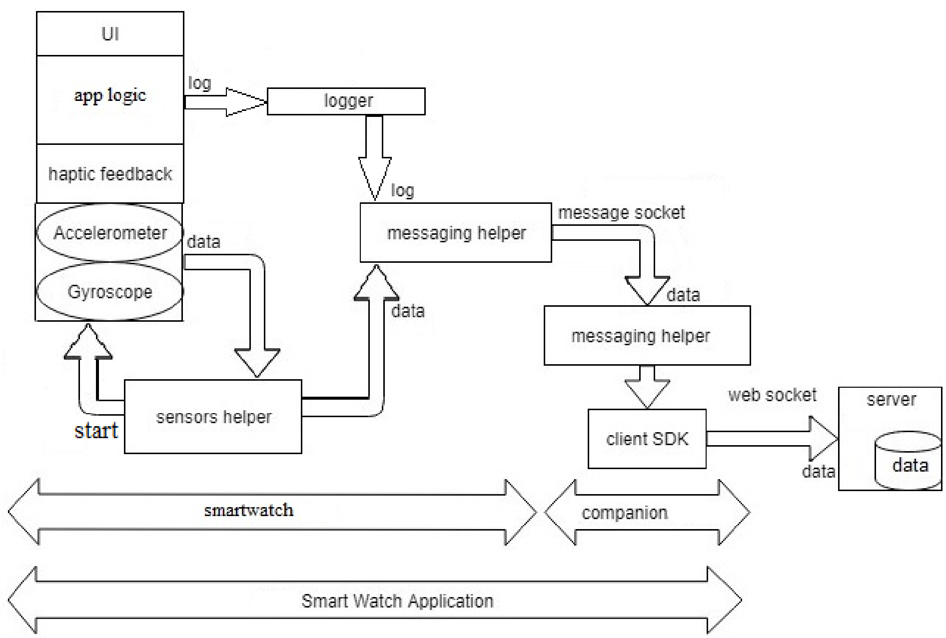


Fig. 2: Data acquisition steps.

- Sensor helper: Sensor helper is the main component of the DCA. This module initiates the sensors, reads out the data, timestamps and buffers them at the same time to handle synchronization between two data streams. Since simultaneous sensor initiation is not guaranteed, the maximum delay between two data streams, according to the sampling rate frequency, is considered in reading and buffering the data. Sensor helper also manages the data buffer size to record as much as data that it is possible. As we were developing an application on a high constraint device, with limited company-implemented functionality, managing the narrow code space was really a challenge. These limitations also resulted in poor data quality because we had to read out and buffer sensors data in a lower frequency than the actual setup-sampling rate. Despite these challenges, we have covered the activity duration completely and sampled as much motion data as it is possible and necessary for the AR process.
- Messaging helper: This module was designed to manage the messaging buffer that can be transferred between smartwatch application and the companion. The mentioned buffer has a limitation of a maximum size of the 1024-byte memory area. Messages with bigger size have to be split into 1024-byte messages and are managed with sequence numbers.
- Logger: Logger is developed to catch the log messages from DCA and transfers them through messaging helper module to be presented in the server-side console.
- Client SDK: Client SDK module communicates with the server through a web API and sends the data file to the server using the HTTP protocol.

The data collecting procedure is shown in Fig.3. In the

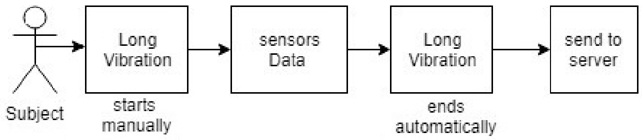


Fig. 3: Data collection procedure.

beginning, the player (subject) wears the watch on the right-hand wrist and gets ready to perform the strokes including serve, backhand, and forehand with a standard tennis racket. When the first long vibration ends; subject starts the activity (tennis stroke) and gets back to the starting position; waits for the second vibration to end. At this moment, data gathering procedure ends automatically and a stop signal is sent to the sensors. Activity data file containing acceleration and angular velocity is provided and by pushing *send to server* button, data file transfer starts.

B. Data pre-processing and activity recognition

As the previous works in tennis AR have utilized dedicated-customized IMUs to collect high-quality data, they did not need to do comprehensive data pre-processing to reach better data quality. However, with the smartwatch, refinement procedures and utilizing data pre-processing techniques are necessary steps to take.

It is worth mentioning some critical issues about the smartwatch that result in poor data quality and nonuniform data streams. Most of these issues are related to hardware and software designs and are out of the developer's control. First, the smartwatch processor does not work in a dedicated way to read out sensors data and it has to handle other processes as well. Second, because of memory shortage, the data streams have to be buffered in a lower frequency than the actual sampling rate and then have to be written to the peripheral storage, which interrupts data streaming and causes considerable loss of data points. Third, the smartwatch run-time environment has a garbage collector routine that starts to run now and then. This also causes interrupts in reading sensors data procedure. In the following, we cover our proposed pre-processing details for smartwatch to reduce the effects of poor data quality in the AR process.

- Data cleansing: From the moment the subject pushes the *start* button, sensors are triggered and data buffering begins, but the early data are irrelevant and should be removed from the dataset. The first buffer flushing is recognized in the datastream by the first noticeable difference between timestamps.
- Alignment: Datastream alignment is performed since the subjects do not perform the activities in constant time duration. Aligning dataset to equal time duration has a positive effect on data segmentation. The data streams will be divided into an equal number of segments. The alignment algorithm finds the maximum duration among streams and zero-pads the streams with a shorter duration.

	Features
1	mean
2	std (standard deviation)
3	median
4	iqr (interquartile range)
5	max
6	min
7	range
8	Percentile 25
9	Percentile 50
10	Percentile 75

TABLE I: Selected time domain features

- Signal processing: Intrinsically, inertial sensors data are very noisy. To reduce the effect of noise, a high-pass, and low-pass filtering is utilized according to the algorithm in [20], in the time domain. A low-pass filter in time domain works as a smoothing function on a datastream by reducing small noises effects. On the other hand, the high-pass filter reduces the gravity effect of the earth on the data, thus more clear data streams are obtained.

The result of the pre-processing procedure is an 18 dimension data matrix. We utilized a fix time-based duration sliding window with overlap for data segmentation. Inertial sensors are continuously sampling the activities, but there is a specific part of the data stream, that contains the effects of the action of interests related data. Motions in sports activities are performing much faster than the daily activities [5] and defining activity boundaries is really a challenge [21]. Additionally, sensor data is highly fluctuated over time, this makes it impossible to classify data over a long period of time [22], [23]. The sliding window is a famous technique in human AR cases [24]–[26]. We considered a 300ms-sliding window for the smartwatch and a 1000ms-sliding window for UTD-MHAD datasets, with 20% overlap for both datasets.

To detect the part of the data that contains the action of interest, we need specific numeric measures. In other words, we need a well-defined feature vector to detect strokes in data streams. The list of time domain features that are extracted for each window or frame of data is listed in Tbl.I, where the range is the difference between the maximum and minimum values in the data vector. These time domain features are selected based on empirical results.

Furthermore, we utilized the feature importance to investigate more effective features in the construction of the activity classification model. The first three important features for all the evaluation scenarios datasets plus their specific data frame number and time duration are listed in Tbl.II. Now, the feature vector is ready as the input for the stroke classification process.

Three shallow supervised machine-learning algorithms are trained and tuned utilizing three-fold cross-validation method. Classifier parameter tuning is done based on exhaustive search

No	Importance Score	Feature	Data dim.	window	Time (msec)
Dataset: Smartwatch 2 strokes					
1	0.3243	mean	Y-R-accl	4	720-1020
2	0.2864	std	Y-LP-gyro	2	240-540
3	0.2672	std	Y-R-gyro	5	960-1260
Dataset: UTD-MHAD 2 strokes					
1	0.0596	mean	Z-R-gyro	2	800-1800
2	0.0465	iqr	Y-LP-accl	3	1600-2600
3	0.0452	percentile-75	Y-HP-gyro	2	800-1800
Dataset: Smartwatch 3 strokes					
1	0.0251	max	Z-R-accl	1	0-300
2	0.0219	std	Y-R-gyro	3	480-780
3	0.0215	max	Z-R-gyro	2	240-540
Dataset: UTD-MHAD 27 strokes					
1	0.0075	percentile-75	X-R-accl	4	2400-3400
2	0.0064	max	X-R-accl	4	2400-3400
3	0.0060	mean	X-R-accl	4	2400-3400

TABLE II: Feature importance. R: raw data, LP: low-pass, HP: high-pass, accl: accelerometer, gyro: gyroscope

approach. We used Python kernel to recognize and classify the strokes. The selected classifiers are K-nearest neighbours, linear SVC and Random Forest (as an ensemble method). The selection was according to *Avci* et al. survey [27] and empirical results. We also reviewed classification results with and without the application of PCA for dimension reduction in all three scenarios.

IV. Performance analysis and discussion

In this section, we evaluate and discuss the results of the proposed method on our smartwatch datasets, and UTD-MHAD dataset. We considered this evaluation in three scenarios.

1) Two strokes (serve and forehand) classification
2) Three strokes classification
3) 27-human actions classification

A. Datasets

To evaluate the proposed method, we use two datasets. The first one is our smartwatch dataset and the second one is UTD-MHAD [12].

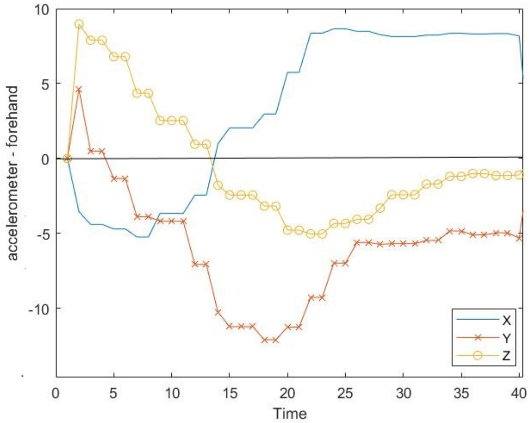
We used Fitbit Ionic™ smartwatch as a wearable sensor to collect inertial motion data. Our smartwatch dataset contains three main tennis strokes of serve, backhand, and forehand. The sampling rate for inertial sensors accelerometer and gyroscope is set to 50Hz. Each subject has repeated every stroke 4 times. Eight right-handed subjects' recordings were selected from the dataset. Therefore, the selected dataset contains 96 data sequences. Data recordings were done under a local area wireless network in the Pervasive Lab at the Shahid Beheshti University [1].

UTD-MHAD dataset [12] contains 27 actions. Eight subjects repeated each action 4 times. After removing three corrupted sequences, the dataset includes 861 data sequences. The dataset is a comprehensive set of human actions covering sports actions, hand gestures, daily activities, training exercises. For hand-related actions, the wearable is placed on a right-hand wrist. Among all these actions, there are two tennis related actions; forehand and serve. For this multi-modal action dataset, one Microsoft Kinect camera and one wearable inertial custom-build sensor were used. The wearable inertial sensor was built in the ESSP Laboratory at the University of Texas at Dallas [28]. It consists of a 9-axis MEMS sensor, which captures 3-axis acceleration, 3-axis angular velocity, and 3-axis magnetic strength with a sampling rate of 50Hz; plus a micro-controller and a dual mode Bluetooth LE. Due to a lack of controlled magnetic field without any distortion in practice, magnetometer data is omitted from the dataset.

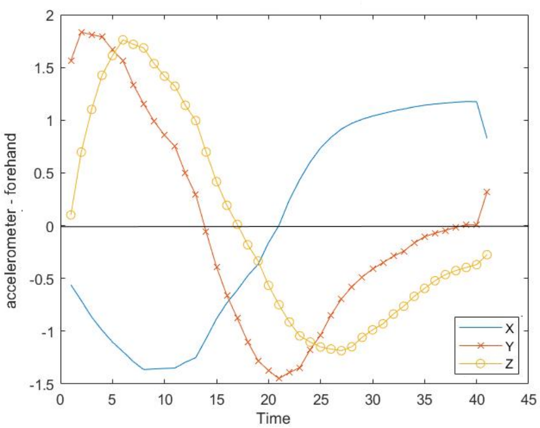
B. Evaluation scenarios

1) 2 tennis strokes evaluation and results: To demonstrate the effects of proposed pre-processing procedures, the waveforms of the accelerometer data for smartwatch forehand stroke are presented. First, the raw data waveform is shown in Fig.4a. The data points in all three axis are scattered and the effects of noise makes the waveform more staircase shaped. After alignment, normalization and low-pass filtering, a smoother waveform with less small noises is obtained

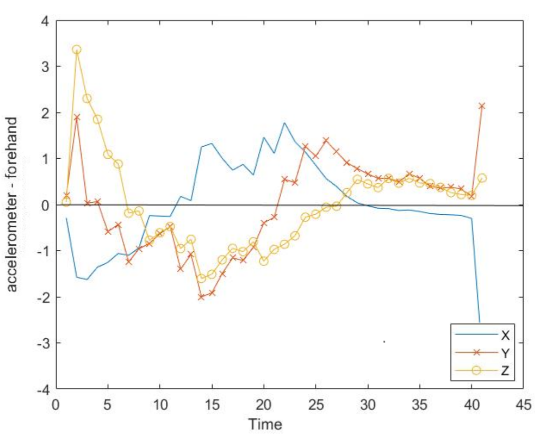
[1]The dataset is publicly available from these sources: http://perlab.sbu.ac.ir/dataset/ https://github.com/sarataghavi/smartwatch-dataset



(a) Raw data waveform.

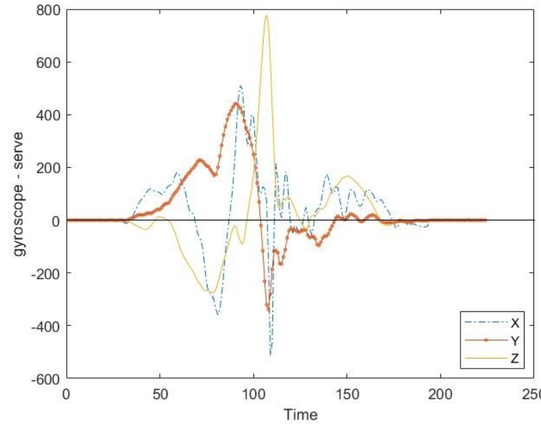


(b) Low-pass waveform.

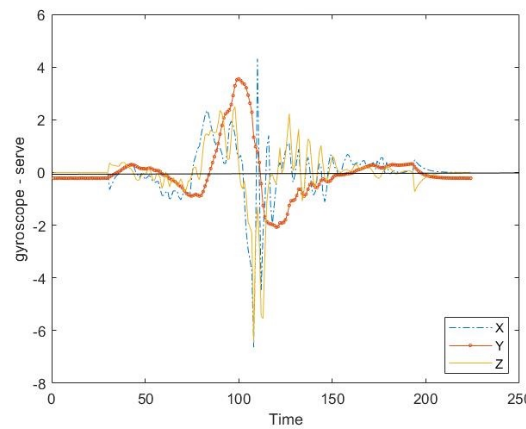


(c) High-pass waveform.

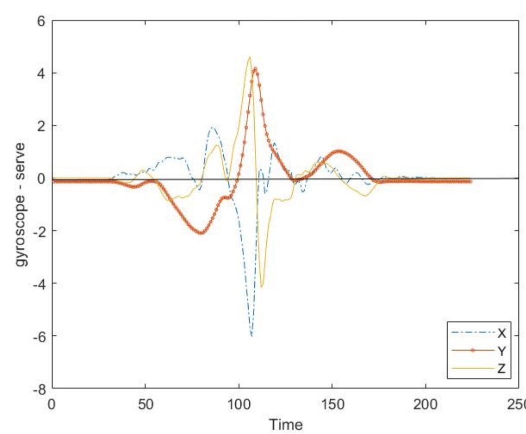
Fig. 4: Smartwatch forehand accelerometer data.



(a) Raw data waveform.



(b) Low-pass waveform.



(c) High-pass waveform.

Fig. 5: UTD-MHAD serve gyroscope data.

(Fig.4b). In addition, there was a significant decrease in wave-form domain from 10 to 2. Fig.4c shows high-pass filtering. This filtration also resulted in waveform domain reduction from 10 to 4, but comparing the shape of the data points with low-pass (LP) filter, the wave-forms are not so smooth.Similar changes observed for smart-watch gyroscope waveform data.

The same pre-processing is performed for the UTD-MHAD 2 strokes sub-dataset. The sample gyroscope wave-forms trans-formation for UTD-MHAD serve stroke is shown in Fig.5. Again a very significant decrease for the wave-form domain happened from 800 to almost 5,after LP filter and we see a less noisy data-points and smoother shape of the wave-form.

Classifier	Parameter tuning & No. of principal components	F1-measure	
		with PCA	without PCA
Random Forest	$n\_estimators = 1$, $max\_depth = 3$ & 1	0.72(± 0.19)	0.70(± 0.10)
Linear SVC	c=0.001 & 22	0.73(± 0.08)	0.73(± 0.08)
KNN	$n\_neighbours$=3 & 15	0.84(± 0.10)	0.81(± 0.08)

TABLE III: classification results for smartwatch two strokes dataset

Classifier	Parameter tuning & No. of principal components	F1-measure	
		with PCA	without PCA
Random Forest	$n\_estimators$=100 , max_depth=4 & 5	1(± 0.00)	1(± 0.00)
Linear SVC	c=0.01 & 2	1(± 0.00)	1(± 0.00)
KNN	$n\_neighbours$=1 & 3	1(± 0.00)	1(± 0.00)

TABLE IV: Classification results for UTD-MHAD two strokes dataset

Classifier	Parameter tuning & No. of principal components	F1-measure	
		with PCA	without PCA
Random Forest	$n\_estimators$=15 , max_depth=10 & 18	0.77(± 0.15)	0.71(± 0.21)
Linear SVC	c=0.001 & 28	0.74(± 0.11)	0.74(± 0.11)
KNN	$n\_neighbours$=3 & 6	0.84(± 0.06)	0.77(± 0.07)

TABLE V: Classification results for three strokes Smartwatch data set

Classifier	Parameter tuning & No. of principal components	F1-measure	
		with PCA	without PCA
Random Forest	$n\_estimators$=1000 , max_depth=25 & 278	0.92(± 0.05)	0.96(± 0.04)
Linear SVC	c=0.01 & 74	0.92(± 0.02)	0.92(± 0.02)
KNN	$n\_neighbours$=3 & 30	0.85(± 0.04)	0.84(± 0.04)

TABLE VI: Classification results for 27-actions in UTD-MHAD

After high-pass (HP) filter a small domain reduction happened specially for axis Z and less noisy data points specially for axis X.

From observing the wave-forms transformations, this can be concluded that LP and HP filters are making better data wave-forms and thus less noisy data-points depending on the performed actions and inertial sensors type and axis. As a result, adding these data-points to the target data-set elevates the quality.

After the data pre-processing, the target data-set for activity detection contains 18 columns. As we have used fix time-based duration sliding window with overlap technique for data segmentation, the feature extraction process result in 1440 features for the smart-watch and 1080 features for the UTD-MHAD datasets.

In the final stage, we used Random Forest, Linear SVC, and KNN classifiers with the two feature vectors. The parameter tuning of the classifiers is performed using *Python scikit-Learn* exhaustive search method *GridSearchcv* with 3-fold-cross-validation for both of the datasets. Then, the best parameter tuning was selected through f1-measure scoring. The parameters that we investigate for each classifier is as follows:

- For the Random Forest the *n_estimator* parameter is the number of trees in the forest and the *max_depth* is the maximum depth of the tree.
- For Linear SVC the *C* parameter is a regularization parameter that controls the trade off between achieving a low training error and a low testing error.
- The parameter *n_neighbours* for kNN is the number of training samples. As the principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these.

The classification results show that the kNN (with k=3 and PCA) was the best classifier for the smart-watch with f-measure of 84%. For UTD-MHAD dataset, all the classifiers reached f-measure of 100%. This implies that the proposed method was successful in elevating the quality of data and boosted the classification results. It also implies that for a

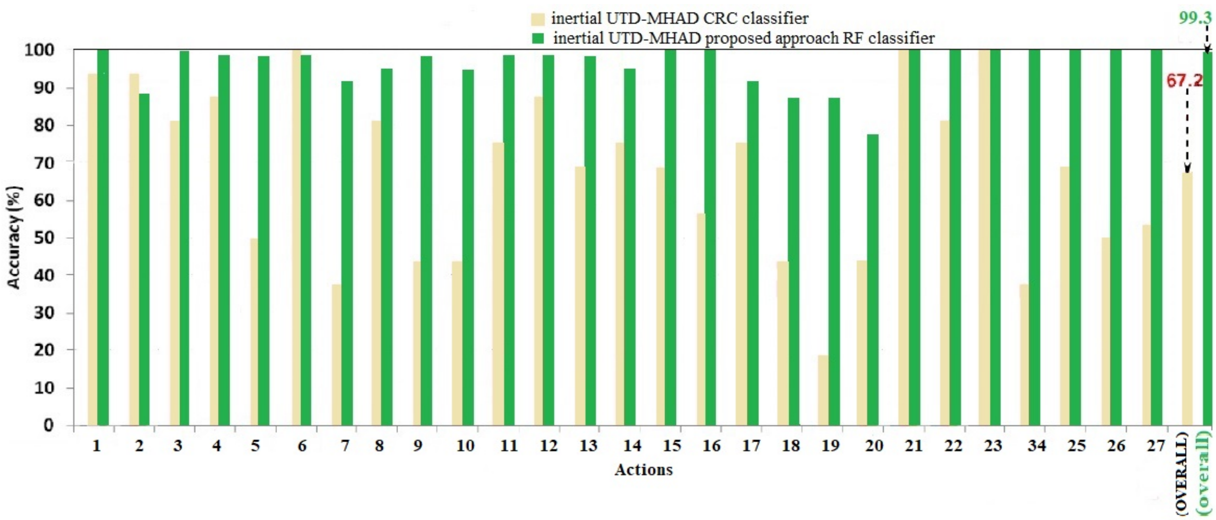


Fig. 6: Classification results for 27-actions in UTD-MHAD.

commercial wearable such as smart-watch , it is necessary to use the proposed method to achieve higher accuracy and precision. The classification results and the parameters are presented in Tbl.III and Tbl. IV.

2) 3 tennis strokes evaluation and results: In this scenario we analyse the method for three main strokes of serve, backhand, and forehand. After pre-processing steps, feature extraction and classification, we observed that all the classifiers had a better performance with PCA. Tbl.V includes the parameter tuning and the number of principal components for each classifier. The 3-NN with PCA and f-measure of 84% was the best classifier in this scenario.

3) The UTD-MHAD dataset includes 27 actions: A full list of the actions can be found in Tbl.VII. The researchers utilized the described method in [29] for inertial feature extraction. That is each acceleration and gyroscope data sequence is partitioned into N=6 temporal windows. The statistical features including mean, variance and standard deviation were calculated for each window. For action recognition, the *Collaborative Representation Classifier* (CRC) described in [30] is used. The regularization parameter λ of the CRC classifier was tuned based on five-fold cross-validation. In the experiments, the data from subjects 1, 3, 5, 7 were used for training, and the data from subjects 2, 4, 6, 8 were used for testing.

We applied the proposed approach to the inertial data of these 27-actions. The same pre-processing and the same features were extracted and fed into the classifiers.

Random Forest classifier without applying PCA on the data reached the highest f-measure of 96%. The detailed results are given in Tbl.VI. Fig.6 reveals the comparison between the two classifications. The Random Forest classifier in the proposed approach reached the overall accuracy of 99.3%, which is 32.1% better than the CRC classifier for the UTD-MHAD for 27 actions dataset.

Wearable inertial sensor on right wrist		
1	right arm swipe to the left	(swipe_left)
2	right arm swipe to the right	(swipe_right)
3	right hand wave	(wave)
4	two hand front clap	(clap)
5	right arm throw	(throw)
6	cross arms in the chest	(arm_cross)
7	basketball shoot	(basketball_shoot)
8	right hand draw x	(draw_x)
9	right hand draw circle (clockwise)	(draw_circle_cw)
10	right hand draw circle (counter clockwise)	(draw_circle_ccw)
11	draw triangle	(draw_triangle)
12	bowling (right hand)	(bowling)
13	front boxing	(boxing)
14	baseball swing from right	(baseball_swing)
15	tennis right hand forehand swing	(tennis_swing)
16	arm curl (two arms)	(arm_curl)
17	tennis serve	(tennis_serve)
18	two hand push	(push)
19	right hand knock on door	(knock)
20	right hand catch an object	(catch)
21	right hand pick up and throw	(pickup_throw)
Wearable inertial sensor on right thigh		
22	jogging in place	(jog)
23	walking in place	(walk)
24	sit to stand	(sit2stand)
25	stand to sit	(stand2sit)
26	forward lunge (left foot forward)	(lunge)
27	squat (two arms stretch out)	(squat)

TABLE VII: Human actions in UTD-MHAD [12]

V. Conclusion

In this work, we proposed an approach for three tennis strokes detection using smartwatch inertial sensors data.

Strokes include serve, backhand and forehand. The system consists of two part. In the first part, a smartwatch data collector application is developed and optimized. Then, the collected smartwatch dataset is used for strokes detection in the second part of the proposed system. For boosting the poor quality of the smartwatch data, some effective pre-processing steps along with PCA dimension reduction method were taken to improve classification performance. We also investigated valuable features by feature importance option. The proposed approach is evaluated in three different scenarios with two datasets; smartwatch and UTD-MHAD datasets. The overall results show a very good improvement in classification accuracy by more than 30%.

REFERENCES

[1] A. Ahmed, M. Asawal, M. J. Khan, and H. M. Cheema, "A wearable wireless sensor for real time validation of bowling action in cricket," in *2015 IEEE 12th Int. Conf. on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2015, pp. 1–5.

[2] A. Tonoli, R. Galluzzi, E. C. Zenerino, and D. Boero, "Fall identification in rock climbing using wearable device," *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*, vol. 230, no. 3, pp. 171–179, 2016.

[3] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Systems with Applications*, 2019.

[4] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[5] P. Esfahani and H. Tabatabaee Malazi, "Pams: A new position-aware multi-sensor dataset for human activity recognition using smartphones," in *2017 19th Int. Symposium on Computer Architecture and Digital Systems (CADS)*. IEEE, 2017, pp. 1–7.

[6] T. Kautz, B. H. Groh, J. Hannink, U. Jensen, H. Strubberg, and B. M. Eskofier, "Activity recognition in beach volleyball using a deep convolutional neural network," *Data Mining and Knowledge Discovery*, vol. 31, no. 6, pp. 1678–1705, 2017.

[7] R. Delgado-Gonzalo, A. Lemkaddem, P. Renevey, E. M. Calvo, M. Lemay, K. Cox, D. Ashby, J. Willardson, and M. Bertschi, "Real-time monitoring of swimming performance," in *2016 38th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2016, pp. 4743–4746.

[8] M. Kos and I. Kramberger, "A wearable device and system for movement and biometric data acquisition for sports applications," *IEEE Access*, vol. 5, pp. 6411–6420, 2017.

[9] M. Zok, "Inertial sensors are changing the games," in *2014 Int. Symposium on Inertial Sensors and Systems (ISISS)*. IEEE, 2014, pp. 1–3.

[10] L. Büthe, U. Blanke, H. Capkevics, and G. Tröster, "A wearable sensing system for timing analysis in tennis," in *2016 IEEE 13th Int. Conf. on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2016, pp. 43–48.

[11] W. Pei, J. Wang, X. Xu, Z. Wu, and X. Du, "An embedded 6-axis sensor based recognition for tennis stroke," in *2017 IEEE Int. Conf. on Consumer Electronics (ICCE)*. IEEE, 2017, pp. 55–58.

[12] C. Chen, R. Jafari, and N. Kehtarnavaz, "Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," in *2015 IEEE Int. Conf. on image processing (ICIP)*. IEEE, 2015, pp. 168–172.

[13] V. Ahanathapillai, J. D. Amor, Z. Goodwin, and C. J. James, "Preliminary study on activity monitoring using an android smart-watch," *Healthcare technology letters*, vol. 2, no. 1, pp. 34–39, 2015.

[14] M. Kos, J. Ženko, D. Vlaj, and I. Kramberger, "Tennis stroke detection and classification using miniature wearable imu device," in *2016 Int. Conf. on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 2016, pp. 1–4.

[15] "Babolat PLAY official webpage," https://en.babolatplay.com/play, accessed: 2019-07-05.

[16] M. Archana and A. Ayyasamy, "Tracking based event detection of singles broadcast tennis video," in *2018 3rd Int. Conf. on Communication and Electronics Systems (ICCES)*. IEEE, 2018, pp. 835–838.

[17] F. Yan, J. Kittler, D. Windridge, W. Christmas, K. Mikolajczyk, S. Cox, and Q. Huang, "Automatic annotation of tennis games: An integration of audio, vision, and learning," *Image and Vision Computing*, vol. 32, no. 11, pp. 896–903, 2014.

[18] "PlaySight (2017, 20 november), smartcourt," https://www.playsight.com/#/tennis, accessed: 2019-07-13.

[19] R. Dhinesh, S. Preejith, and M. Sivaprakasam, "Tennis serve correction using a performance improvement platform," in *2018 IEEE 6th Int. Conf. on Serious Games and Applications for Health (SeGAH)*. IEEE, 2018, pp. 1–7.

[20] H. Gjoreski and M. Gams, "Accelerometer data preparation for activity recognition," in *Proceedings of the Int. MultiConf. Information Society, Ljubljana, Slovenia*, vol. 1014, 2011, p. 1014.

[21] D. J. Cook and N. C. Krishnan, *Activity learning: discovering, recognizing, and predicting human behavior from sensor data*. John Wiley & Sons, 2015.

[22] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Data preprocessing for supervised leaning," *Int. Journal of Computer Science*, vol. 1, no. 2, pp. 111–117, 2006.

[23] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE communications surveys & tutorials*, vol. 15, no. 3, pp. 1192–1209, 2012.

[24] J. Wu, A. Akbari, R. Grimsley, and R. Jafari, "A decision level fusion and signal analysis technique for activity segmentation and recognition on smart phones," in *Proceedings of the 2018 ACM Int. Joint Conf. and 2018 Int. Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. ACM, 2018, pp. 1571–1578.

[25] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li, "Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations," in *Int. Conf. on ubiquitous intelligence and computing*. Springer, 2010, pp. 548–562.

[26] J.-Y. Yang, J.-S. Wang, and Y.-P. Chen, "Using acceleration measurements for activity recognition: An effective learning algorithm for constructing neural classifiers," *Pattern recognition letters*, vol. 29, no. 16, pp. 2213–2220, 2008.

[27] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga, "Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey," in *23th Int. Conf. on architecture of computing systems 2010*. VDE, 2010, pp. 1–10.

[28] C. Chen, K. Liu, R. Jafari, and N. Kehtarnavaz, "Home-based senior fitness test measurement system using collaborative inertial and depth sensors," in *2014 36th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2014, pp. 4135–4138.

[29] C. Chen, R. Jafari, and N. Kehtarnavaz, "Improving human action recognition using fusion of depth camera and inertial sensors," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 1, pp. 51–61, 2014.

[30] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *2011 Int. Conf. on computer vision*. IEEE, 2011, pp. 471–478.