Smart Health 16 (2020) 100114

Contents lists available at ScienceDirect

Smart Health

journal homepage: www.elsevier.com/locate/smhl

When your wearables become your fitness mate

Xiaonan Guo [a,*], Jian Liu [b], Yingying Chen [b]

[a] Indiana University-Purdue University Indianapolis, IN, 46202, USA
[b] WINLAB, Rutgers University, NJ, 08902, USA

ARTICLE INFO

Keywords:
Mobile health
Wearable devices
Exercise recognition
Exercise review and recommendation
Gesture control

ABSTRACT

Acknowledging the powerful sensors on wearables and smartphones enabling various applications to improve users' life styles and qualities (e.g., sleep monitoring and running rhythm tracking), this paper takes one step forward developing FitCoach, a virtual fitness coach leveraging users' wearable mobile devices (including wrist-worn wearables and arm-mounted smartphones) to assess dynamic postures (movement patterns & positions) in workouts. FitCoach aims to help the user to achieve effective workout and prevent injury by dynamically depicting the short-term and long-term picture of a user's workout based on various sensors in wearable mobile devices. In particular, FitCoach recognizes different types of exercises and interprets fine-grained fitness data (i.e., motion strength and speed) to an easy-to-understand exercise review score, which provides a comprehensive workout performance evaluation and recommendation. Our system further enables contactless device control during workouts (e.g., gesture to pick up an incoming call) through distinguishing customized gestures from regular exercise movement. In addition, FitCoach has the ability to align the sensor readings from wearable devices to the human coordinate system, ensuring the accuracy and robustness of the system. Extensive experiments with over 5000 repetitions of 12 types of exercises involve 12 participants doing both anaerobic and aerobic exercises in indoors as well as outdoors. Our results demonstrate that FitCoach can provide meaningful review and recommendations to users by accurately measure their workout performance and achieve 93% and 90% accuracy for workout analysis and customized control gesture recognition, respectively.

1. Introduction

There is clear evidence that lack of exercise is linked to the rise in obesity (Ladabaum, Mannalithara, Myer, & Singh, 2014), and that regular exercise can prevent and treat obesity and its metabolic complications (Bouchard, Depres, & Tremblay, 1993). Thus, to keep a healthy lifestyle, people have been encouraged to visit gym regularly and hire personal trainers to customize exercise regimens, track compliance with those regimens, and perfect form during exercise. However, these options present a number of barriers: people who work find it difficult to squeeze in time to go to a dedicated fitness facility to exercise; there are associated costs in gym membership and hiring trainers. Therefore, an increasing number of people are choosing to exercise in their home or at the gym without hiring trainers. But the downside is also obvious: once the trainer/coach is not available to provide help and advise on posture and form, the possibility of injury due to incorrect exercise form increases. This can be a very important issue, especially for individuals who are overweight or suffer from obesity, as they are more prone to injury during exercise. In addition, people also would like to record their exercise details to

* Corresponding author.
E-mail addresses: xg6@iupui.edu (X. Guo), jianliu@winlab.rutgers.edu (J. Liu), yingche@scarletmail.rutgers.edu (Y. Chen).

https://doi.org/10.1016/j.smhl.2020.100114
Received 30 June 2017; Received in revised form 13 June 2019; Accepted 6 February 2020
Available online 11 February 2020

keep track of their fitness plan. For example, a user performs various exercises such as free-weight with dumbbells, stretch, and running on a treadmill, the corresponding exercise statistics could help to infer meaningful health-related information (e.g., calories burned). Therefore, in this paper, we aim to develop a smart exercise assistant system that facilitates exercise monitoring and assessment.

The proliferation of *wearable mobile devices* (e.g., smartwatches, wrist-worn fitness bands, and smartphones mounted on arms) has the potential to provide fitness assistant because it showed initial success on improving our life styles through a great number of applications in smart healthcare, smart home, and smart cities. An important use case of wearable mobile devices is providing guidelines to improve people's daily activities, for example, tracking walking steps (Ren, Chen, Chuah, & Yang, 2013a), monitoring sleep qualities (Liu et al., 2015), and estimating daily caloric intake (Mifflin et al., 1990). In this work, we take one step forward by answering the question: whether such wearable mobile devices become powerful enough leveraging fine-grained sensing information to perform systematic comprehensive fitness assistance and prevent injuries.

Traditionally, fitness monitoring is performed by analyzing the workout captured by video tapes (Arney, 2005) or specialized sensors (Chang, Chen, & Canny, 2007; Cheng et al., 2013). Chang et al. (Chang et al., 2007) track free-weight exercises by incorporating an accelerometer into a workout glove. Cheng et al. (Cheng et al., 2013) develop a technique that can recognize human activities by attaching a sensor on users' hips. In recent years, smartphone apps and fitness trackers, such as Sworkit (Sworkit), FitStar (Fitstar), Jefit (Jefit) and Flyfit (Fitbit), show the initial success of fitness monitoring without using additional equipments (e.g., cameras or specialized sensors). They can perform step counts and log exercises based on users' manual inputs. These apps require explicit inputs from users, including the type of workout and the start/stop time. Hao et al. (Hao, Xing, & Zhou, 2015) present a system using smartphone and its external microphone that detects running rhythm and improves exercise efficiency for runners, yet the question whether or not mobile devices can automatically distinguish different types of exercises and provide fine-grained performance recommendation related to exercises remains open.

Toward this end, we take one step forward to search for an integrated mobile solution that can perform systematic fitness monitoring and performance review as well as facilitate in-exercises device control. We propose *FitCoach* leveraging wearable mobile devices to achieve the following three main aspects: **(i) Fine-grained Fitness Data Interpretation.** Recording the sensor readings on wearable mobile devices (e.g., smartwatch or smartphone) during workout to explore their capability of deriving fine-grained exercise information including exercise types, the number of sets and the number of repetitions (reps) per set. The derived quantitative data can be further analyzed for inferring meaningful information (e.g., calories burned). **(ii) Smart Exercise Guidance.** Furthermore, the derived fitness data is of great importance to assist the users to maintain proper exercise postures and avoid injuries. To build muscles and gain a healthier body, it is widely recognized that people should perform their workout properly and effectively. FitCoach aims to not only regulate the workouts by following the Frequency, Intensity, Time and Type (FITT) principle (of Sports Medicineet al., 2013), but also provide detailed guidelines to review the user's posture through workout and provide recommendation in keeping correct exercise form (e.g., in terms of speed of exercise execution and strength). **(iii) In-exercise Contactless Device Control.** There is an increasing demand on controlling mobile devices (e.g., taking an incoming call or controlling music playing) during workout. FitCoach designs customized gestures and distinguishes them from regular exercise activities to perform contactless device control without user stopping or impeding movements.

In particular, FitCoach exploits *Short Time Energy* (STE) to derive fine-grained fitness data (i.e., strength and speed of body movements) in exercises and recognize different types of exercises automatically by using embedded sensors (e.g., accelerameter and gyroscope) on wearable mobile devices. Rooted in the understanding of body movements in exercises, FitCoach develops a novel metric for evaluating the quality of each user's exercises, *exercise form score*. This exercise form score reflects the difference of strength and speed of body movements between each repetition of an exercise based on a reference profile. The reference profile could be either obtained from the user's own sensor data or built from other people's data (e.g., training coaches or members from the same fitness club) through crowdsourcing platforms (e.g., fitness club's facebook, WhatsApp or WeChat).

The contributions of our work are summarized as follows:

- Assessing dynamic postures (movement patterns & positions) automatically during workout including anaerobic as well as aerobic exercises.
- Achieving fine-grained exercise recognition (including exercise types, the number of sets and repetitions) without user involvement.
- Calculating exercise form score and providing performance review to evaluate the performed workout and prevent injuries.
- Enabling contactless device control during workout through distinguishing customized gestures from regular exercise movements (e.g., arm swing during jogging), requiring no assumption with respect to the status of users (e.g., moving or standing).
- Aligning sensing data into the human coordinate system to ensure high recognition accuracy and achieve system robustness even when the real-time data possess the different device facing direction or exercise direction comparing to the reference profiles.
- Evaluating the system performance involving 12 people using both smartwatches and mobile phones in armbands during both gym and outdoor workouts, which can achieve over 90% high accuracy for both workout analysis and customized control gesture recognition.

The rest of the paper is organized as follows. We review related studies in Section 2. In Section 3, we describe challenges of developing FitCoach and present the system design. We then introduce techniques that achieve exercise recognition and workout review in Section 4 and Section 5, respectively. Next, we describe how to achieve accurate control gesture recognition during exercise in Section 6. In Section 7, we present the implementation of FitCoach. We evaluate FitCoach in Section 8. Finally, we discuss the open issues and conclude our work in Sections 9 and 10 respectively.

2. Related work

Recent studies show that life experience can be improved through implementing various types of techniques using sensors and wireless technologies including gesture recognition (Abdelnasser, Youssef, & Harras, 2015; Doliotis, Stefan, McMurrough, Eckhard, & Athitsos, 2011; Kurakin, Zhang, & Liu, 2012; Park et al., 2011; Pu, Gupta, Gollakota, & Patel, 2013; Ren et al., 2013b; Suk, Sin, & Lee, 2010), activity recognition (Cheng et al., 2013; Keally, Zhou, Xing, Wu, & Pyles, 2011; Mokaya, Lucas, Noh, & Zhang, 2015; Parate, Chiu, Chadowitz, Ganesan, & Kalogerakis, 2014; Thomaz, Essa, & Abowd, 2015; Vlasic et al., 2007; Wang et al., 2014) and physical exercises monitoring (Hao et al., 2015; Mokaya et al., 2015; Sundholm, Cheng, Zhou, Sethi, & Lukowicz, 2014; ; Chang et al., 2007).

Gesture recognition serves as a communication interface between human and machines. For example, video-based technologies can capture and recognize human hand motion (Doliotis et al., 2011; Kurakin et al., 2012; Ren et al., 2013b; Suk et al., 2010) but require line-of-sight. Alternatively, gestures can be recognized by leveraging WiFi signals (Abdelnasser et al., 2015; Pu et al., 2013). These solutions can be affected by people nearby and the movement of other body parts. Recently, Park et al. (Park et al., 2011) use a hand-worn sensor device and a smartphone for gesture recognition during exercises. It can only recognize a limited number of gestures because their mechanism does not take device orientation into consideration and cannot differentiate similar gestures performed at different directions.

Another aspect of related studies focuses on activity recognition including daily activities (Cheng et al., 2013; Keally et al., 2011; Mokaya et al., 2015; Vlasic et al., 2007) and healthcare related activities such as eating (Thomaz et al., 2015) and smoking (Parate et al., 2014). Vlasic et al. (Vlasic et al., 2007) develop a full body motion capture system by using multiple sensors attached on a human body. Similarly, Keally et al. (Keally et al., 2011) combine on-body sensors and a smartphone to recognize activity. Cheng et al. (Cheng et al., 2013) develop a technique that can recognize activities without training by placing a sensor on users' hips. These studies show that either external sensors or sensors embedded in wearables have the capability to accurately recognize human daily activities. Wang et al. (Wang et al., 2014) propose to leverage WiFi with location information to recognize activities but their solution is vulnerable under multi-person scenarios.

Automatically monitoring physical exercises have attracted more attention recently. There are mobile Apps (Sworkit; Fitstar; Jefit.), wristband (Fitbit) and solutions based on mobile devices with sensors (Chang et al., 2007; Hao et al., 2015; Mokaya et al., 2015; Sundholm et al., 2014). Chang et al. (Chang et al., 2007) propose to track free weight exercises by incorporating an accelerometer into a workout glove. Mokaya et al. (Mokaya et al., 2015) utilize multiple sensors attached to human body to monitor muscle activation. In addition, Sundholm et al. (Sundholm et al., 2014) use sensor matrix as a mat to recognize gym exercises. Along this line, Hao et al. (Hao et al., 2015) propose to monitor the running rhythm by measuring breathing and strides with headsets and smartphones. These techniques rely on additional sensors or specific hardware. Most importantly, whether a workout feedback and guidance can be further provided to improve exercise performance is still an open question.

In contrast, FitCoach only uses wearable mobile devices (e.g., wrist-worn smartwatches or arm-mounted smartphones) to provide fine-grained tracking of workout and further offer exercise review and guidance to improve fitness experience. In addition, a control gesture recognition functionality is integrated into FitCoach, and it is robust regardless of device orientation.

3. Design of FitCoach

3.1. Challenges and practical issues

Realizing FitCoach, a virtual coach that can provide guidelines and convenience during physical exercises, using sensors in a single wearable mobile device faces a number of challenges and practical issues:

Exercise Form Correction Using Single Wearable Mobile Device. In order to allow a system to provide exercise form correction, it is necessary for the system to understand the performance of an exercise through the body movements performed during the exercise, which is a very challenging task to cope with by using a single wearable mobile device. This is because commercial mobile devices usually have limited low-power sensing modalities (i.e., accelerometers, gyroscopes, and magnetometer). It is even more challenging when it comes to the single-device solution, because the device is attached to one point of the body and only captures partial information of the body movements of the exercises. Therefore, the system needs to be designed in such a way that it can provide exercise form corrections based on the dynamics of sensor data resulted from the partial knowledge of the exercises.

Robust Fine-grained Exercise Differentiation. It is also very challenging to utilize sensors in wearable mobile devices to correctly distinguish so many different types of exercises that are often performed in gyms, because sensor readings collected from the wearable mobile devices are extremely noisy due to the dynamic nature of exercises. Thus, it is important to devise a robust exercise classifier that can eliminate the impact of unstationary noises in the sensor data and capture the fine-grained differences between different types of exercises.

Gesture Control during Exercises. Allowing users to perform device control during exercises based on customized gestures is attractive but also very hard. Basically, the dynamics of the user are dominated by significant body movements of exercises, which can be considered as strong "noises" interfering with the signal patterns in the sensor data caused by the customized gestures. Therefore, we need to design an intelligent method to detect and separate the signals generated by customized gestures from the "noises" generated by body movements of exercises.

Automated Wearing Orientation Alignment. During exercises, wearable mobile devices may change its facing from the original direction from time to time. Such orientation changes result in unstable projection of user's body movements in the mobile device's coordinate system, and makes it hard for the system to determine the pattern of body movements. Therefore, the system needs to include

a light-weight alignment algorithm that can automatically transform the sensor data collected under unstable orientation to that referring to a stable orientation to facilitate accurate exercise recognition and gesture control during exercises.

3.2. System overview

Nowadays, people indulge themselves in using mobile devices around the clock. We notice that more and more people tend to wear their mobile devices during workout for either entertainment (e.g., listening to music or reading messages) or functionality (e.g., making telephone calling or recording running trajectories), which stimulates the desire of using mobile devices to automatically recognize different types of exercise and monitor their qualities. The main goal of FitCoach is to examine the users' dynamics (i.e., body movement patterns & intensities) in workouts and provide detailed workout statistics to assist users to achieve effective workouts and prevent injuries.

As illustrated in Fig. 1, the repetitive pattern of body movements in exercises can be well captured by using the inertial sensors of the wearable mobile device. It can extract fine-grained fitness information (e.g., basic statistics, motion strength and performing period) and provide illustrative feedback to users, which can also be exploited to enforce the Frequency, Intensity, Time, Type (FITT) principle of training (of Sports Medicineet al., 2013).

As illustrated in Fig. 2, FitCoach takes as input time-series of sensor readings including acceleration, gyroscope and quaternion, all of which are readily available in off-the-shelf wearable mobile devices. The system first performs *Workout Detection* to filter out sensor readings that don't contain workout activities based on the presence of periodicity pattern in workout activity. The sensor readings that are found to contain workout activities will be served to three tasks, *Workout Interpretation &Recognition*, *Workout Review/Recommendation* and *Customized Gesture Recognition*. The Workout Recognition performs quantitative analysis to the sensor readings and identifies different types of workouts based on the acceleration features that can capture unique repetitive patterns of different exercises. The Workout Review/Recommendation examines the characteristics of each rep (i.e., strength and time intervals) and provides two exercise form scores as feedback to users for performance evaluation. Meanwhile, the *Customized Gesture Recognition* identifies pre-defined gestures based on their unique patterns in sensor readings to facilitate contactless device control during workouts.

Particularly, the *Workout Interpretation &Recognition* consists of four major components: *Earth-reference Coordinate Alignment*, *Set/Rep Counting and Segmentation*, *Accel-based Feature Extraction*, and *Exercise Classification*. The *Earth-reference Coordinate Alignment* tackles the

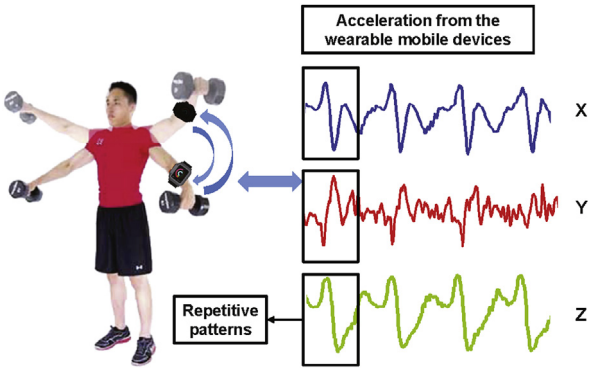


Fig. 1. Movement in exercises can be revealed by repetitive patterns of sensor readings from wearable mobile devices.

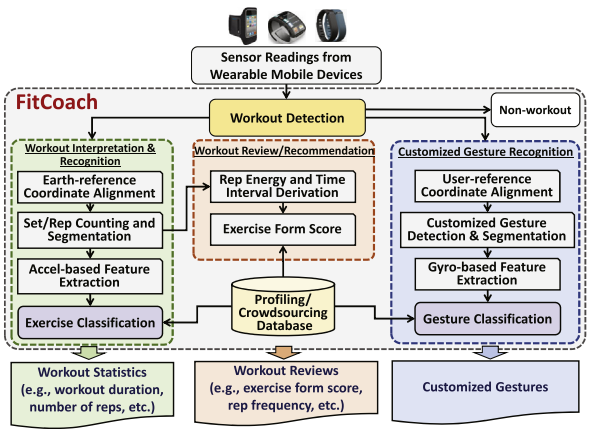


Fig. 2. FitCoach framework.

issue of dynamic orientation in workouts, and automatically rotates sensor readings to a fixed world coordinate system. Then FitCoach counts the number of sets during workout and the number of reps in each set based on the magnitude of the repetitive signals. The sensor readings are further divided into small segments corresponding to the detected reps. In each segment, the *Accel-based Feature Extraction* derives statistics features that capture each repetitive moving patterns of exercises from three-axis acceleration readings. After workout interpretation, the system performs *Exercise Classification*, which utilizes a profile based algorithm to determine the types of exercises by comparing the extracted features with those of pre-collected profiles in the *Profiling/Crowdsourcing Database*.

In addition, the *Workout Review/Recommendation* aims to provide systematic fitness monitoring and performance review as feedback to users, which would assist the users to maintain proper exercise gestures and avoid injuries. FitCoach takes the segments of sensor readings identified in the *Set/Rep Counting and Segmentation* as inputs, and performs the *Rep Strength and Time Interval Derivation* to estimate the characteristics of body movements in exercises (i.e., strength and frequency of the repetitive motions). The estimated characteristics are further utilized by the *Exercise Form Score Calculation* to calculate the exercise form score for each rep, which is a novel metric that allows the users to easily understand their performance in the exercises.

Furthermore, FitCoach performs *Customized Gesture Recognition* to distinguish pre-defined customized gestures from regular exercise activities, which would facilitate the increasingly demanded in-exercise device control. In order to achieve accurate gesture recognition in exercises, the system first utilizes the *User-reference Coordinate Alignment* to deal with the dynamically changing orientation of the sensing devices during exercises by rotating the sensor readings to the user's coordinate system. Then the system performs the *Customized Gesture Detection & Segmentation* to extract the segments of data that contains the gestures. The segments are further utilized to derive unique gesture-related features from the gyroscope readings in the *Gyro-based Feature Extraction*. At last the system adopts a profile based classifier in the *Gesture Classification* to determine customized gestures by comparing the derived gyroscope features with those stored in the profiling/crowdsourcing database.

4. Workout interpretation & recognition

4.1. Workout detection

A key observation is that most regular exercises involve repetitive arm movements. For example, jogging and walking involve periodic arm swing, and weight lifting involves periodic pushing-ups. Such repetitive arm movements result in regularly changing values in sensor readings. In addition, the repetitive patterns from exercises tend to last for a long time period simply because people normally adopt a set-and-rep scheme in exercise to maximize the effectiveness. Compared to regular exercises, non-workout activities usually don't have such long-term repetitive pattern. Therefore, we propose to detect workout based on determining whether there are long-term repetitive patterns in the sensor readings.

Towards this end, our system adopts an autocorrelation-based approach to examine the accelerations resulted from exercise motions. The autocorrelation approach is a common technique used for detecting repetitive patterns in a time series (Parthasarathy, Mehta, & Srinivasan, 2006). In particular, FitCoach first applies a moving time window with the length of w to the time series of accelerometer readings. For each time window, the system uses the Magnitude of Linear Acceleration (*MLA*) to estimate the linear acceleration (i.e., acceleration without gravitational acceleration) of exercise motions, which allows the system to detect the repetitive patterns worrying about the direction of the exercise motions. The MLA based on accelerometer readings can be derived by the following equation:

$$MLA(i) = \sqrt{(a(i)_x)^2 + (a(i)_y)^2 + (a(i)_z)^2} - g, \tag{1}$$

where $a(i)_x, a(i)_y$ and $a(i)_z$ are the acceleration of the ith sample on the x, y and z axis of the mobile device respectively and g is the acceleration of gravity. Note that, the MLA in Equation (1) equals to zero when there is no motion.

Then, FitCoach calculates the autocorrelation of the time series of MLA, and use a typical peak finding algorithm (Liu et al., 2015) to find the number of peaks, which is denoted as N_p. The number of detected repetitive patterns thus can be derived with $N_r = (N_p - 1)/2$, due to the symmetric nature of the autocorrelation. Finally, to accommodate the noisy accelerometer readings, we use a threshold-based method to confirm the detected repetitive patterns are resulted from workouts. The workout detection results for each window can be derived by:

$$D_w = \begin{cases} 1, & N_r > \nu \\ 0, & otherwise, \end{cases} \tag{2}$$

where D_w is a boolean value depicts whether the given sensor readings within a window belong to workout or not. D_w outputs 1 when N_r is bigger than a threshold value ν. Fig. 3 shows an example of our workout detection results with $w = 5$s and $\nu = 3$, which demonstrates that our system can accurately detect the windows containing workouts.

4.2. Set/rep segmentation

Once workout-related activities have been detected, FitCoach will perform coordination alignment because when people are doing exercise, the sensor readings from a mobile device are usually defined in the device coordinates. We introduce Quaternion-based coordinate alignment, which converts sensor reading from device coordinate to human coordinate. The details can be found in Section 7.

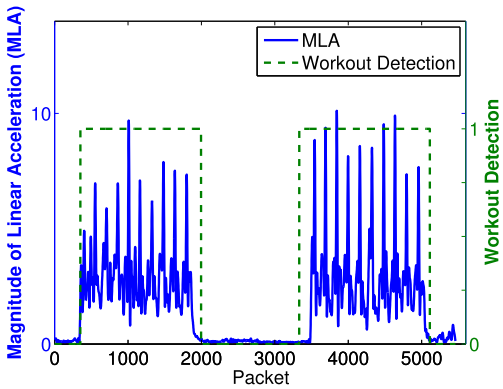


Fig. 3. Workout detection based on a 5-s sliding window (output 1 if the number of repetitive patterns is larger than 3 within the window, otherwise output 0).

After coordinate alignment, FitCoach integrates the windows that are continuously labeled as workouts into a segment, which corresponds to a set of repetitive activities for any type of workouts. The time intervals between any two segments are identified as the rest intervals, which will be provided as a part of the exercise review and is also an important contributor to the success of any strength training program. However, in order to provide fine-grained exercise performance information, FitCoach needs to look into the data in each set and analyzes the data based on a finer-grained concept, *repetition/rep*.

To further determine the segment of reps in a set, we devise a motion-strength-oriented approach to accurately determine the starting and ending time point of each repetition within a set. The intuition behind the approach is that each repetition usually consists of a series of arm movements that results in a unique pattern in terms of the accumulated motion energy: 1) the accumulated energy starts to increase sharply from zero when the arm moves from an initial position to an ending position; 2) the accumulated energy drops a little when the arm pauses at the ending position for a very short while; 3) the accumulated energy starts to increase sharply again when the arm moves back from the ending position to the initial position; and 4) finally the accumulated energy drops sharply when the hand stops at the initial position for some rest. We found that this unique pattern of accumulated motion energy can be captured by the wearable mobile device through the Short Time Energy of MLA. Fig. 4 illustrates the relationship of the unique pattern in the accumulated energy and the arm movements in each repetition.

Particularly, we adopt the Short Time Energy (STE) (Deller, Proakis, & Hansen, 1993) to capture the unique energy pattern in the time series of MLA derived from Equation (1). The basic idea of this step is to accumulate the energy of the MLA in short sliding windows, which is defined as following equation:

$$E_{sqr} = \sum_{i=m}^{n} \left[V(i)^2\right]. \tag{3}$$

We only consider the sensor data that within a frame and thus, Equation (3) is written as,

$$E(n) = \sum_{m}^{n} \left[MLA(i) \cdot W(n-i)\right]^2, \tag{4}$$

where $MLA(i)$ is the MLA value of the ith sensor reading, n is the number of samples at which we compute the STE, $W(n)$ represents the window function of finite duration and we use rectangular window function for its simplicity. After obtaining STE of MLA, FitCoach

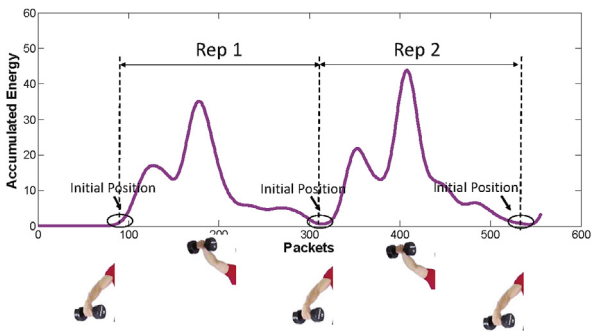


Fig. 4. Illustration of the relationship between the arm movements in a repetition and the unique pattern of accumulated energy captured by a wearable mobile device (i.e., a smartphone in an armband).

applies the same peak finding algorithm used in Section 4.1 to detect the peaks in STE. Then the system finds the local minimum point between two peaks as the ending point of each repetition, and the data between two detected ending points are defined as a repetition segment. Fig. 5 shows an example of determining the repetition segments based on the local minimum points that are detected in STE of MLA from a wearable mobile device (i.e., a smartwatch) when a user conducts 15 repetitions. The results indicate that the motion-strength-based approach can accurately separate the data for each repetition.

4.3. Accel-based feature extraction & workout classification

After repetition segmentation, FitCoach aims to identify the workout type for each set. The basic idea is to build a database with the profiles for different types of workouts before the workout classification, then we use a profile-based approach to determine the workout type for each rep segment in a set, and further to infer the workout type of the entire set. Next, we focus on the feature extraction and workout classification. We will discuss how to construct the database of profiles in Section 7.3.

Accel-based Feature Extraction. In order to distinguish workout types, we need to find the features that capture the unique characteristics of each workout type. Based on our extensive feature selection studies, we finally determine nine statistical acceleration-based features that are most useful to distinguish different types of workouts, namely *skewness, kurtosis, standard deviation, variance, most frequently appear in the array, median, range, trimmean* (Doane and Seward) *and mean.* To extract features without worrying about the variation of the mobile device's orientation, we first perform the world-reference alignment to rotate all acceleration data to the world coordinate system. The details of the world-reference alignment are provided in Section 7.1. After the world-reference alignment, FitCoach extracts the nine acceleration-based features from the already aligned three-axis accelerations in each rep segment to describe the body movements. In total, we extract 27 features (i.e., nine features per axis) for each rep segment.

Light-weight Classifier. FitCoach utilizes a light-weight machine learning based approach to identify different types of workouts based on the acceleration-based features extracted from each rep segment. It is light-weight because the system only needs to determine the workout type for the first few rep segments within a set, and the workout type of the entire set of repetitions is identified as the majority decision based on the classification results for the first few rep segments. Specifically, we adopt a Support Vector Machine (SVM) (Vapnik & Vapnik, 1998) classifier with radial basis function kernel (Schölkopf et al., 1997). The classifier is trained based on the pre-collected profiles of different types of workouts, which is described in Section 7.3.

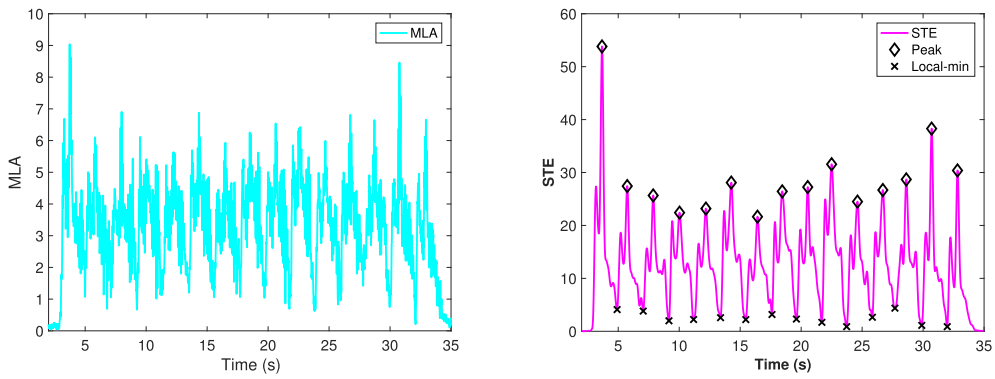
5. Workout review and recommendation

In this section, we first sketch the big picture of the workout review provided by FitCoach through summarizing the workout statistics provided by the system. Then, we discuss the details of our novel exercise form score and workout review plane, which is a unique way to visualize the exercise form score.

5.1. Exercise form score design

Besides providing basic workout statistics to users, FitCoach aims to offer users a more intuitive way to understand their performance in exercises by comparing their exercise statistics to a baseline, which could be either generated based on the users' own data or based on the data from croudsourcing. Towards this end, we define a novel metric named *exercise form score*, which consists of two subscores that respectively evaluate a user's fine-grained performance of each rep in the exercise based on two important criteria as shown below:

Motion Strength (MS). A proper exercise form should maintain the motion strength at a certain level. For example, too much strength may indicate that the weight is too heavy and increases the risk of injury while too little strength may indicate that the weight is



(a) Magnitude of Linear Acceleration (MLA)

(b) Local minimum identified in Short Time Energy (STE) of MLA

Fig. 5. Example of rep segmentation for 15 repetitions of dumbbell raise exercise.

too light to build muscle effectively. We intuitively utilize the strength level of each rep to describe the motion strength, which means a set of reps with good performance should maintain a stable strength level. The strength level of each rep can be estimated by the maximum value in obtained STE of MLA.

Performing Period (PP). A proper exercise form should avoid too-fast or too-slow movements in order to effectively build muscles and prevent injuries. In this work, we utilize the time period of each rep to describe the performing period of each rep, which reflects how fast a user performs a repetition in exercises. Therefore, a set of reps with good performance should also have similar time periods. The time period of each rep can be directly obtained from the length of each rep segment after the segmentation described in Section 4.2. We note that the performing period provides more insights to users. For example, users can leverage such information for equipment weight adjustment (e.g., reduced speed of last few reps in a set indicates that the user may be training exhausted and need to decrease the weight or number of reps in next set).

Exercise Form Score. Based on these two criteria, FitCoach defines the *Exercise Form Score*, which consists of two subscores: *MS score* and *PP score*. The subscores depicts how the testing rep deviates from the baseline in terms of the motion strength and performing period, respectively. We discuss the details about the baseline in the next subsection. Particularly, the MS score for the ith rep is defined as:

$$E_i = \frac{A(i) - A^*}{A^*}, i = 1, 2, 3, \ldots, n, \tag{5}$$

where $A(i)$ is the maximum STE of the MLA of the ith rep, and A^* is the motion strength baseline. Similarly, the PP score for the ith rep is defined as:

$$T_i = \frac{I(i) - I^*}{I^*}, i = 1, 2, 3, \ldots, n, \tag{6}$$

where I_i is the length of the ith rep and I^* is the performing period baseline. The output exercise form score is a 2-tuple score that can be denoted as $< E_i, T_i >$.

5.2. Personal/Crowdsourcing Baseline

The exercise form score reflects the performance of the testing rep comparing to a baseline. We design two baselines that are suitable in different scenarios, namely *Personal Baseline* and *Crowdsouring Baseline.*

Personal Baseline. We observe that users usually can perform exercises with standard strength and frequency at the beginning of the workout, but the quality of the exercises decays with time due to fatigue. Based on this observation, a good candidate of the baseline for evaluating the performance of a user's workouts is the early portion of the user's own reps. In particular, we derive the personal baseline by averaging the motion strength and performing period of the first k reps of the first set in the user's sensor data. We empirically choose $k = 5$ in our work.

Crowdsourcing Baseline. The personal baseline is good for short-term exercise performance evaluation but could bias to the user's own preference. For example, a user could feel tired at the beginning of the exercise and result in bad baseline for evaluating the entire exercise. To tackle this problem, we further propose the crowdsourcing baseline, which allows users to compare their performance with the baseline from exemplars (e.g., fitness coaches, bodybuilders, and amateur expertise) to achieve a long-term and more accurate exercise performance evaluation. The crowdsourcing approach is feasible because it is an increasing trend that people would like to share their fitness data in online social network to earn credits or build record, and more social platforms, such as WhatsApp and WeChat, start to provide the functionality allowing people to share their fitness data among friends.

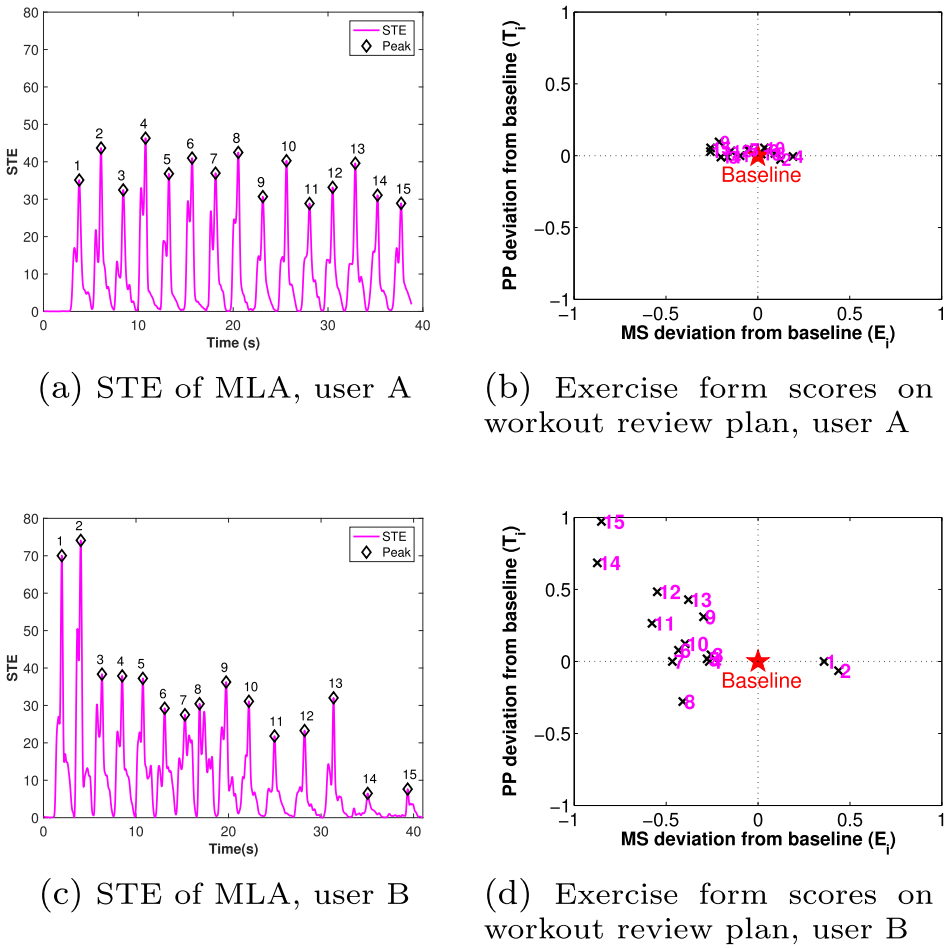
5.3. Workout review plane

FitCoach further adopts a unique view angle of the exercise form score to allow users to track the performance of their each rep in an illustrative way. In particular, we define a *review plane* in which the x axis and y axis are the MS score and PP score, respectively. According to Equations (5) and (6), the Origin represents the rep having the exactly same performance as the chosen baseline, and every exercise form score $< E_i, T_i >$ corresponding to the ith rep can be mapped to a position in the review plane. Apparently, the rep having its position closer to the Origin has better performance, and the more reps close to the Origin the better.

Fig. 6 compares the workout reviews of two different users (i.e., User A an User B) in a set of lateral raising exercises (i.e., 15 reps in one set). Fig. 6(a) and (c) respectively depict STE of MLA of two users' reps, which show that User A has more stable strength levels and time lengths for each repetition than User B. Fig. 6(b) and (d) respectively illustrate two users' exercise form scores based on their personal baselines in the review planes, which shows that the score points of User A are concentrated around the Origin while the score points of User B are scattered around the second quadrant of the review plane. The observation indicates that User B has much higher motion strength and longer performing period comparing to the user's first few reps, and thus has worse performance than User A.

6. Customized gesture recognition

Next, we discuss how to recognize a user's customized gestures during rest time between two sets of exercises or during aerobic



(a) STE of MLA, user A

(b) Exercise form scores on the workout review plan, user A

(c) STE of MLA, user B

(d) Exercise form scores on the workout review plan, user B

Fig. 6. Comparison of the Short Time Energy (STE) of the Magnitude of Linear Acceleration (MLA) and the exercise form scores on the workout review plane between user A and user B.

exercises (e.g., walking and jogging). In such case, the orientation of the wearable mobile device may be always changing. FitCoach thus adopts a user-reference coordinate alignment to convert the sensor readings obtained from the mobile device to the coordinate system defined based on the user's facing direction to make sure the system can capture the unique sensor reading patterns associated with the body movements regardless the device's orientation. We leave the details of user-reference coordinate alignment and facing direction estimation in Section 7.

6.1. *Customized gesture detection & segmentation*

From sensor point of view, both accelerometer and gyroscope can reflect the performed gestures. However, accelerometer is more sensitive to motions, therefore the steps from jogging can cause acceleration changed dramatically. To mitigate the impact of steps, we resort to gyroscope readings in our customized gesture recognition.

In order to achieve customized gesture controlling while doing aerobic exercises (e.g., jogging), we need to differentiate the controlling gesture from regular exercise motions such as arm swing while running. We observe that such regular motions are only confined in some small space. For example, arm swing is always along with the same direction as heading (i.e., facing direction) which shows significant gyroscope readings in x axis in human coordinate as shown in Fig. 7 where four controlling gestures are performed while running. Thus we could further reduce the impacts from these regular arm swing motions by ignoring the readings of x axis in human coordinate.

We first compute Magnitude Rotation (MR) from gyroscope readings of the rest two axes: $MR = \sqrt{((g_y)^2 + (g_z)^2)}$, where g_y and g_z are the rotation value from y-axis and z-axis of gyroscope respectively. Then we calculate the short time energy (STE) of MR to further detect and segment each performed customized gestures. As illustrated in Fig. 8, calculated STE of MR has four obvious peaks representing four different customized gestures. After that, we use the similar method of rep segmentation as discussed in Section 4.2 to detect and

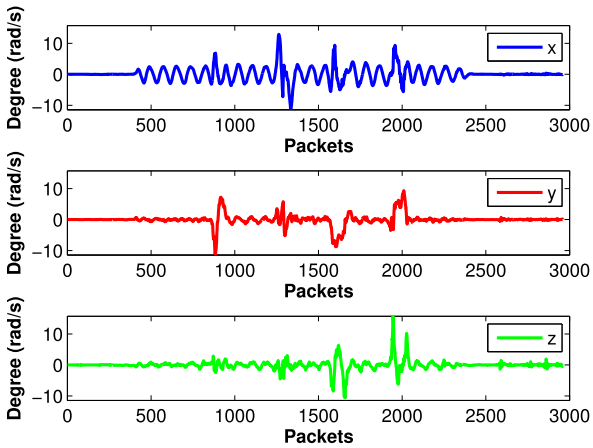


Fig. 7. Gyroscope readings in human coordinate of four different customized gestures while running.

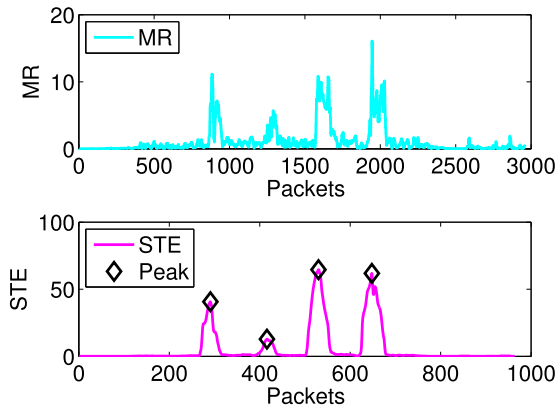


Fig. 8. Magnitude rotation (MR) and its corresponding STE with peaks associated with the presence of customized gestures.

segment each customized controlling gesture.

6.2. Gyro-based feature extraction & gesture classification

To distinguish different customized gestures, we use the similar features as we extracted in exercise classification which is discussed in Section 4.3. Different from accel-based feature extraction, we only use *y* and *z* axes in gyroscope readings. Nine statistical features that are extracted from the gyroscope readings of these two axes, which include *skewness, kurtosis, standard deviation, variance, most frequently appear in the array, median, range, trimmean* (Doane and Seward) *and mean*. Note that gyroscope readings have been aligned to human coordinate before the feature extraction.

For the gesture classification, we adopt the same classifier as we use in the exercise classification, which is Support Vector Machine (SVM) (Vapnik & Vapnik, 1998) classifier with radial basis function kernel (Schölkopf et al., 1997). FitCoach classifies the performed customized gestures based on its extracted gyro-based features. The classifier is trained by the pre-collected profiles of different types of gestures in the profile database, which is described in Section 7.3.

6.3. Design philosophy of customized gestures

In order to well differentiate the performed gestures from regular exercise motions such as arm swing while running, we need to elaborately design the customized gestures. Given that regular motions will result in significant gyroscope amplitude of *x* axis in human coordinate, we need to design customized gestures that only produce great changes of gyroscope readings on *y* and *z* axes. It means the gestures we designed should be mainly within the surface that is perpendicular to the human's facing direction. For example, we can define gesture “lift the arm to the right” which generates significant gyroscope readings mainly on *y* axis in human coordinate.

7. Implementation

7.1. Quaternion-based coordinate alignment

During exercises, users wearing wearable mobile devices basically involves three different coordinate systems as illustrated in Fig. 9, namely, *mobile device coordinate*, *earth coordinate*, and *human coordinate*. The sensor readings from a mobile device are usually defined in the device coordinate and thus result in non-fixed projection of the user's body movements defined in the human coordinate. This makes it hard for the system to determine the patterns of body movements during exercises directly from the sensor readings. In order to address this issue, FitCoach adopts quaternion (Quaternion and https://en.wi) based approaches to dynamically convert sensor readings from the mobile device coordinate either to the human coordinate or to a coordinate system having the fixed mapping to the human coordinate.

7.1.1. Earth-reference alignment

For exercise recognition in a gym, both the earth and human coordinates are fixed since the user's facing orientations are restricted by exercise machines or areas. Therefore, our system only needs to convert sensor readings from the mobile device coordinate to the earth coordinate, where the projection of body movements during exercises is fixed regardless the wearable mobile device's orientation.

Specifically, we convert the sensor readings from the mobile device coordinate to the earth coordinate by using the quaternion-based rotation $p_e = q_{me} p_m q_{me}^{-1}$, where p_m is the sensor reading vector (e.g., accelerations) in the mobile device coordinate, and q_{me} is the quaternion reading from the mobile device coordinate to the earth coordinate, which can be obtained from the device directly. q_{me}^{-1} is the conjugate quaternion of q_{me}. After conversion, the converted sensor readings p_e are in the earth coordinate and can provide stable patterns of body movements during exercises to enable our exercise recognition discussed in Section 4.3.

7.1.2. User-reference alignment

For customized gesture recognition, we notice that most people only use gesture control during the rest time between two sets of exercises in the gym or during aerobic exercises (e.g., walking and jogging). In such cases, users' facing orientations may change and the mapping between the earth coordinate and the human coordinate is no longer fixed. Therefore, our system has to convert the sensor readings in the mobile device coordinate to the human coordinate.

Specifically, we convert the sensor readings from the mobile device coordinate to the earth coordinate by using the quaternion-based rotation $p_h = q_{mh} p_m q_{mh}^{-1}$, where p_m and p_h are the sensor reading vector (e.g., rotation rates) in the mobile device coordinate and the human coordinate respectively. q_{mh}^{-1} is the conjugate quaternion of q_{mh}, and q_{mh} is the quaternion readings from the mobile device to the human coordinate, which can be calculated using Hamilton product: $q_{mh} = q_{he}^{-1} q_{me}$, where q_{me} is the quaternion reading from the mobile device coordinate to the earth coordinate, which can be obtained from the device directly. q_{he}^{-1} is the conjugate quaternion of q_{he}, and q_{he} is the quaternion readings from the human to the earth coordinate, which can be derived from the estimated facing direction.

More specifically, we can derive $q_{he} = [w, x, y, z]$ using the *Euler angles* in earth coordinate which is defined as:

$$\begin{cases} w = cos\left(\frac{\varphi}{2}\right) cos\left(\frac{\theta}{2}\right) cos\left(\frac{\psi}{2}\right) - sin\left(\frac{\varphi}{2}\right) sin\left(\frac{\theta}{2}\right) sin\left(\frac{\psi}{2}\right); \\ x = cos\left(\frac{\varphi}{2}\right) sin\left(\frac{\theta}{2}\right) cos\left(\frac{\psi}{2}\right) + sin\left(\frac{\varphi}{2}\right) cos\left(\frac{\theta}{2}\right) cos\left(\frac{\psi}{2}\right); \\ y = cos\left(\frac{\varphi}{2}\right) sin\left(\frac{\theta}{2}\right) cos\left(\frac{\psi}{2}\right) - sin\left(\frac{\varphi}{2}\right) cos\left(\frac{\theta}{2}\right) sin\left(\frac{\psi}{2}\right); \\ z = cos\left(\frac{\varphi}{2}\right) cos\left(\frac{\theta}{2}\right) sin\left(\frac{\psi}{2}\right) + sin\left(\frac{\varphi}{2}\right) sin\left(\frac{\theta}{2}\right) cos\left(\frac{\psi}{2}\right); \end{cases} \tag{7}$$

where rotation angles φ, θ and ψ are the *row*, *pitch* and *yaw* (i.e., counterclockwise rotation angle of x, y, z-axis respectively) respect to earth reference respectively as shown in Fig. 9. We assume that people running on the horizontal ground and therefore φ and θ are equal to zero and we only need to calculate facing direction ψ (i.e., *yaw*) which can be estimated through arm swing while running from

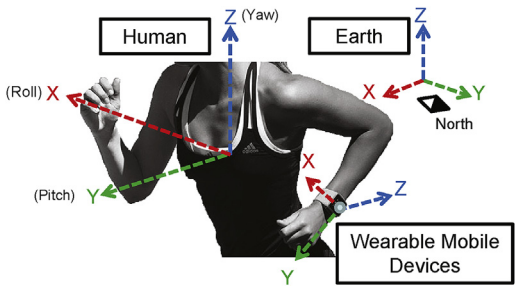


Fig. 9. Three coordinate systems.

equation (8) in Section 6.

7.2. Facing direction estimation

It is hard to directly measure the facing direction merely using mobile devices. We then resort to indirect measurement. In particular, we exploit the arm swing direction to estimate the user's facing direction. The key insight is that arm swing in human walking is a natural motion wherein each arm swings with the motion of the opposing leg. Swinging arms are in an opposing direction with respect to the lower limb. Therefore, it is feasible to estimate facing direction based on the arm swing direction. In particular, FitCoach segments each arm swing using rep segmentation as described in Section 4.2, then converts the acceleration readings from the mobile device's coordinate into the earth coordinate as discussed in Section 7.1.1. After conversion, we can double integrate the acceleration projected to the x and y axes in the earth coordinate to derive the moving distance of the arm along the x and y axes, respectively. In this work, we define the arm swing direction as the counter-clockwise rotation around the z-axis from y-axis in the earth coordinate (i.e., North direction), which is similar to the definition of *yaw* in *Euler angles*. We first calculate the included angle δ between the displacement of x-axis and y-axis caused by arm swing by using $\delta = \left|arctan(s_y/s_x)\right|$,where s_x, s_y are the distance accumulated from acceleration in *x*-axis and *y*-axis respectively by using *Trapezoidal rule* (Lindberg, 1971). Note that δ is ranging from 0° to 90° and then we need to convert it from 0° to 360°. Therefore, we need to decide the quadrant *Q* of arm swing direction, that is defined in Cartesian system where *x* and *y* are East and North in earth reference respectively, to convert it to ψ ranging from 0° to 360° as:

$$\psi = \begin{cases} 270+\circ\delta; & if\ Q = 1, \\ 90-\circ\delta; & if\ Q = 2, \\ 90+\circ\delta; & if\ Q = 3, \\ 270-\circ\delta; & if\ Q = 4, \end{cases} \tag{8}$$

where *Q* can be determined based on the order of maximum and minimum values (i.e., peak and valley) on *x* and *y* axes of accelerometer readings. For example, when the arm swing moves toward $\delta = 45^\circ$, and the peak appears before the trough in both *x* and *y* axes of acceleration both, then $Q = 1$, while the peak appears after the trough on the *x* axis and experiences the opposite on the *y* axis indicating $Q = 2$.

We also conduct an experiment to evaluate the proposed facing direction estimation. In the experiment, a volunteer is asked to run toward four different directions (i.e., north, south, east and west in earth reference). Fig. 10 shows the 10-round estimation results for each direction. We find that the estimated directions are along with the four running directions, and the little bias is caused by the fact that people swing their arms naturally while running which is not perfectly stuck to their facing directions.

7.3. Profiling database construction

When users start FitCoach for the first time, they are asked to build a profiling database for the exercise recognition and customized gesture recognition by performing the particular types of exercises and customized gestures. FitCoach extracts the accl-based and gyro-based features as discussed in Section 4.3 and 6.2, and asks the user to manually label the features with corresponding exercise types and customized gestures. We note that FitCoach allows users to wear the wearable mobile devices at flexible positions when constructing the

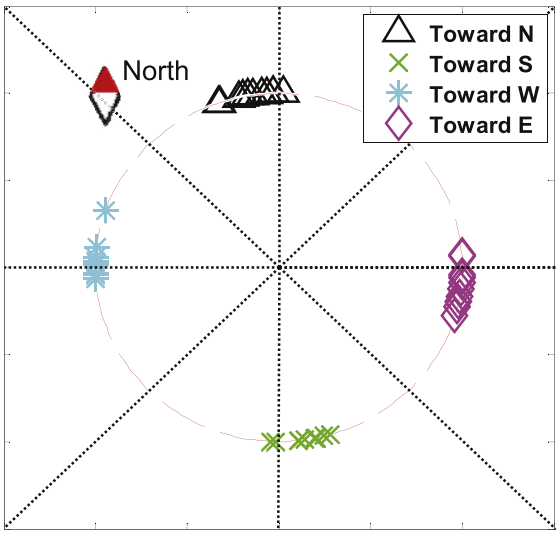


Fig. 10. Facing direction estimation of four running directions: toward North (N), South (S), West (W) and East (E).

profiling database, because the quaternion-based coordinate alignment always converts sensor readings to a coordinate system that has the fixed mapping relationship to the human coordinate during exercises.

7.4. User interface

FitCoach currently provides two main services to the users: fitness monitoring and workout review. In particular, fitness monitoring automatically tracks the performed exercises in terms of the number of sets, the number of repetitions per set and the exercise type. This service aims to help users to record workout statistics. In addition, workout review provides feedback information to users in terms of two exercise form score that we defined in Section 5.1. These two scores can help the users to adjust their following exercise form. Fig. 11 shows the user interface of FitCoach.

8. Performance evaluation

In this section, we first present the experimental methodology and metrics we used to evaluate FitCoach. We then evaluate the performance and robustness of FitCoach using both smartwatch and smartphone during people's fitness workout.

8.1. Experimental methodology

8.1.1. Wearable mobile devices

We evaluate FitCoach with two types of wearable mobile devices (i.e., a smartphone of Samsung Galaxy Note 3 and a smartwatch of LG Watch Urbane) to study the impact of devices and the wearing positions. Both devices use Android and can collect sensor readings of accelerometer, gyroscope and quaternion vector. In our experiment, the participants are asked to wear the smartwatch on the wrist with

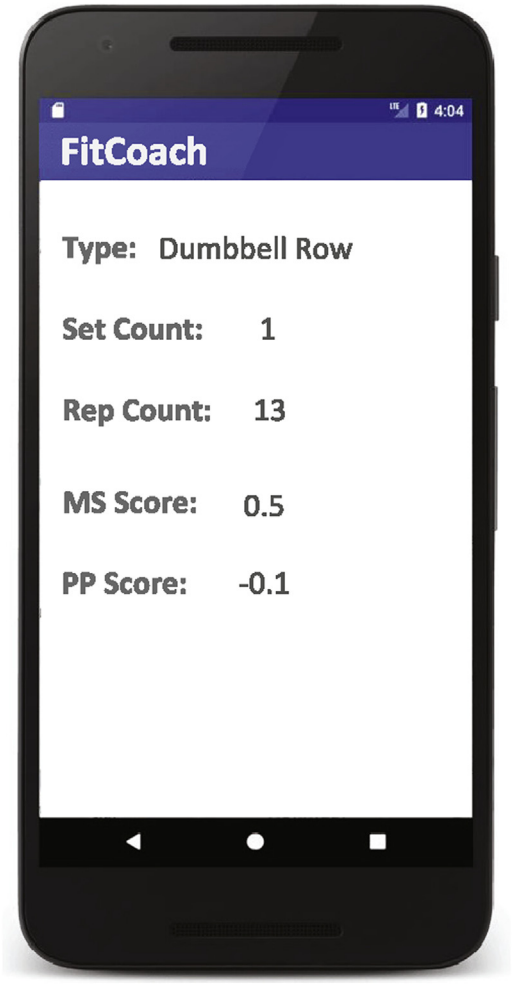


Fig. 11. FitCoach user interface.

their own wearing preferences and the phone is mounted on their upper arms using a jogging armband. During exercise, acceleration, gyroscope and quaternion are collected with the sampling rate of 100 Hz. The ground truth of workout statistics are recorded by a volunteer as an observer.

8.1.2. Fitness data collection

We conduct extensive experiments using both smartwatch and smartphone. We recruit 12 volunteers from colleagues, friends and students from research lab. Among them, 7 out of 12 go to gym regularly and the rest go to gym less frequently. For over a half year experiments, all 12 volunteers are asked to wear the smartwatch and smartphone simultaneously at the same arm, which is for the performance comparison between smartwatch and smartphone of the same exercises. In addition, a volunteer accompanies with them as well to record the ground truth of their workout statistics. Specifically, we study 12 different exercise types, as illustrated in Fig. 12, which are usually seen in gym including both aerobic and anaerobic exercise with or without machine. In total we collect over 5000 repetitions of 12 types of exercises involving 12 participants. The tested exercises include both anaerobic exercises, including weight machines and free weights, and aerobic exercises in which around 2 h running is tested in both indoors (e.g., treadmill) and outdoors.

In order to evaluate customized gesture recognition, participants are also asked to randomly perform four pre-defined customized gestures while running. According to the design philosophy of customized gestures as we discussed in Section 6.3, we defined four different customized gestures as shown in Fig. 13. We use these four gestures as examples and evaluate our system accordingly, but FitCoach also applies to more gestures. The four defined customized gestures include: (1) lift the arm to the right; (2) lift up the arm to the top; (3) rotate the arm in front of body clockwise; and (4) rotate the arm in front of body anticlockwise. In total, over 500 customized gestures are performed by 12 participants.

8.2. Evaluation metrics

We use the following metrics to evaluate the performance of FitCoach:

Precision. Given N_e reps of an exercise/gesture type e in our collected data, precision of recognizing the exercise type e is defined as $Precision_e = N_e^T/(N_e^T + M_e^F)$, where N_e^T is the number of instances collectedly recognized as exercise e. M_e^F is the number of sets corresponding to other exercises that mistakenly recognized as exercise e.

Recall. Recall of the exercise type e is defined as the ratio of the reps that are correctly recognized as the exercise e over all reps of exercise type e, which is defined as $Recall_e = N_e^T/N_e$.

F1-score. F1-score is the harmonic mean of precision and recall, which reaches its best value at 1 and worst at 0. In our multi-class scenario, the F1-score for a specific gesture e was defined as $F_1^{(e)} = 2 \times \frac{precision_e \times recall_e}{precision_e + recall_e}$.

Rep/Gesture Detection Rate. Given all reps of an exercise type e, rep detection rate is defined as the ratio of the number of detected reps of e over all reps of e the user performed. Gesture detection rate is defined as the ratio of the number of customized gesture detected

Fig. 12. Illustration of 12 types of exercises.[11]

Fig. 13. Illustration of four different customized gestures.

for type c over the number of type c gestures performed by users.

8.3. Workout recognition using smartwatch

We first evaluate the performance of FitCoach on workout exercise recognition using smartwatch. Fig. 14(a) shows the confusion matrix of the recognizing exercise types by using smartwatch in FitCoach. In the confusion matrix, the numbers are shown in percentage. The rows of the confusion matrix are the actual exercises the users performed (i.e., ground truth) and the columns are recognized exercises. An entry M_{ij} denotes the ratio between the number of exercise i was predicted as gesture j and the number of the total number of i. The average accuracy is 95% with standard deviation 5% over all 12 types of exercises. We find that recognizing results from $E1$ and $E10$ are relatively low, which are 85% and 89% respectively. This may be caused by some volunteers who go to the gym less frequently and cannot maintain the exercise in a correct form for all reps. For example, $E10$ (i.e., Dumbbell Biceps Curl) is free weight exercise and some volunteers may not maintain their arm within a fixed space all the time. For exercise $E1$ (i.e., Barbell Bench Press), some volunteers easily perform too fast or too slow depending on the weights.

In addition, Fig. 14(b) presents the precision, recall and F_1 score for each exercise type, respectively. The average value of precision, recall and F_1 score of each exercise are all around 95%. Although the recall of exercise $E4$ (i.e., running) is 100%, we observe that it has the lowest precision among all 12 exercises, which indicates other exercises are more likely to be mistakenly classified as this exercise. This may be caused by the fact that arm swings are naturally moving in space and some volunteers freely perform some type of exercise too fast which also involve all axes sensor readings. The above results support that FitCoach can extract accurate information for exercise type recognition through wrist-worn smartwatch.

8.4. Workout recognition using smartphone

We then evaluate workout recognition by using smartphone since arm-mounted phone have been widely used in people's daily exercise. We present the results from smartphone in Fig. 14 (c) and Fig. 14 (d). Results show 91% average recognition accuracy for exercise recognition and customized gesture recognition respectively. We find exercise $E4$ still has the lowest precision which is consistent with the results collected from smartwatch since the volunteers wear smartwatch and smartphone on the same arm to make fair comparison.

Comparison between Smartwatch and Smartphone. FitCoach presents high accuracy of workout recognition for both smartphone and smartwatch. Comparing results between smartwatch and smartphone, we found that results obtained from smartwatch are better than results from smartphone. The average recognition accuracy of smartwatch is 95% whereas smartphone has a 91% average recognition accuracy. This observation is due to the fact that for exercise recognition, the space scope of the arm gesture trajectories was constrained by the machine for some exercise and most of the exercises require users to use their hands to grab and therefore the smartwatch on the wrist is close to hand and reflect more similar movement as machine or dumbbell.

8.5. Comparison of various classifiers

In this part, we further compare the system performance under four traditional classifiers, including k nearest neighbors (k-NN), decision tree (DT), random forest (RF) and support vector machine (SVM). Each classifier is evaluated using 10-fold cross-validation, and the parameters of each classifier are tuned to achieve the best performance. Fig. 16 shows the workout recognition accuracy of four different classifiers for smartwatch and smartphone respectively. We observe that SVM achieves the best performance among the four classifiers. Specifically, for smartwatch, the four classifiers have average accuracies of 91%, 78%, 83%, 95%, respectively. And for smartphone, the four classification models have average accuracies of 86%, 82%, 83%, 91%, respectively. Moreover, we find SVM has the lowers variance of 4% and 5% for smartwatch and smartphone respectively. The result shows that SVM generates better results than other three classifiers for our application scenario.

8.6. Workout review

Our system provides workout review to each user by comparing a user's workout pattern to the baseline in terms of the two metrics

[1] By courtesy of app *Fitness Buddy*.

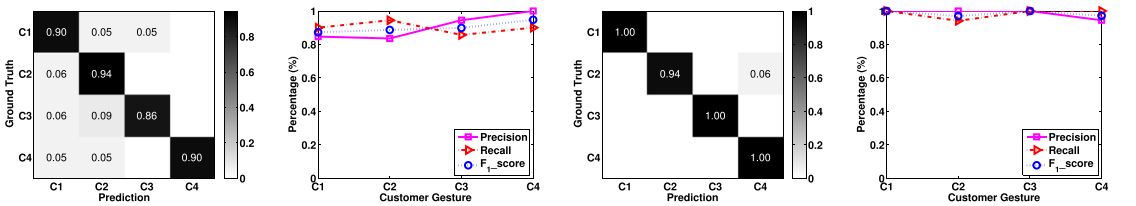
defined in Section 5. Then, users can directly review how well they perform each repetition and adjust the following repetition according to the workout review plane. During this project, all the volunteers try our workout assessment and adjust their postures based on the workout review plane provided by our system. In particular, we ask 5 expert users, who visit gym regularly and have training experiences before, to perform three sets of workout with 10 repetitions per set for each workout type as the crowdsourcing baseline. Then, other volunteers can adjust their posture accordingly. Overall, all volunteers give positive feedback on our system and confirm that the proposed system can help the users to correct their exercise form.

8.7. Customized gesture recognition

Next, we show how accurate FitCoach can match the right customized gestures while doing aerobic exercise (i.e., running). The dataset is collected from a total 2-h exercise including 1.5 h treadmill running and 0.5 h outdoor running. Volunteers are asked to perform pre-defined customized gesture randomly with smartwatch and smartphone on the same arm.

The classification confusion matrix of recognizing the four customized gesture using a smartwatch is shown in Fig. 15(a). We observe that the average recognition accuracy is around 90% which reflects FitCoach can achieve high customized gesture matching while doing aerobic exercise. The mistakenly recognized gestures are mainly categorized as the first two gestures. This may be caused by the characteristics of the four gestures design, in which all four movements are mainly rotation-like based gestures. From Fig. 15(b), we find that the average value of the corresponding precision, recall and F_1 score are all around 90% which demonstrates FitCoach can provide accurate gesture-based control to interact with mobile devices while doing exercise. We find that smartphone presents better results as shown in Fig. 15(c) and (d). We observe that the average recognition accuracy is around 98.5% and the average value of the corresponding precision, recall and F_1 score are all around 98%. The reason is that for customized gesture, there is no space restriction of the arm movement trajectories and therefore the sensor readings from smartwatch undergo more fluctuation.

Robustness under Challenging **Scenario.** Given different device orientations caused by rotation of wristband or armband, we have shown that sensor readings can be completely different with two same gestures. In addition, people may change their directions while running. Therefore, traditional approaches can not avoid profiling process based on per-person manner since the profile built from one person can not directly be reused for another person due to the reasons mentioned above. We show that, FitCoach maintains a high recognition accuracy under challenge scenario as shown in Fig. 17. In this set of experiment, 5 out of 12 volunteers participant and test FitCoach in a 5-fold manner: one of the five persons will be served as a tester and the other will build the profile database. Then the tester performs pre-defined customized gesture while running and deliberately wears the smartwatch in a different orientation. Then FitCoach leverages the profiles built from the other four people to recognize the gestures performed by the tester. Through the experiments, we



(a) Confusion matrix, smartwatch (b) Precision/recall/F1 score, smartwatch (c) Confusion matrix, phone (d) Precision/recall/F1 score, phone

Fig. 15. Comparison of the performance of customized gesture recognition between using a smartwatch and a smartphone.

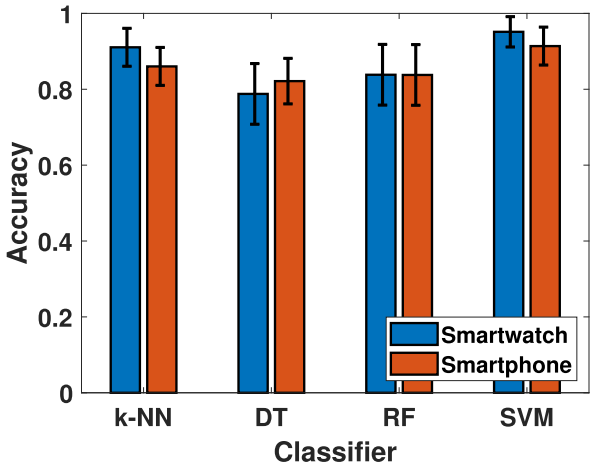


Fig. 16. Comparison of different classifiers.

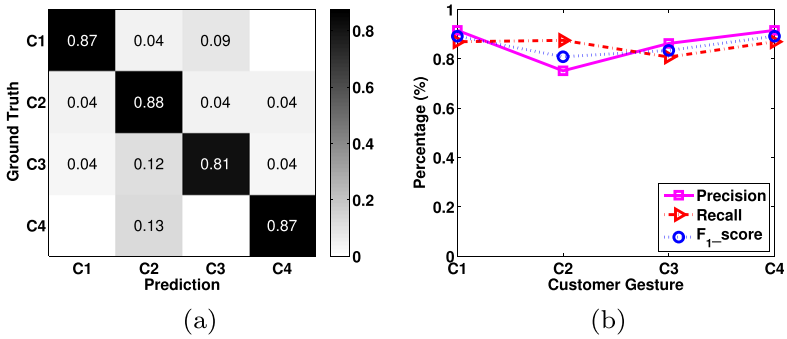


Fig. 17. (a) Confusion matrix of customized gesture by using profile database obtained from others. (b) Corresponding precision, recall and F_1 score.

find that FitCoach still achieves an 86% average recognition accuracy across 5 different people by using the profile from other people directly. The results support the claim that our coordinate alignment algorithm is effective and robust under challenging scenario.

8.8. Rep/gesture detection rate

Finally, we evaluate FitCoach by showing our detection rate for both exercises (for each rep) and customized gestures (for each gesture) by using a smartwatch. For workout exercise detection, the average detection accuracy reaches 99%. The lowest detection rate occurs at running exercise $E4$ (i.e., step detection) on a treadmill but it still achieves around 95% detection accuracy as shown in Fig. 18(a). Such relative low detection rate of running exercise is cased by occasionally holding on the handrails or wiping perspiration while running. Fig. 18(b) presents the results for customized gestures where FitCoach successfully detects all the presences of workout reps. The above results show that FitCoach can accurately detect reps as well as customized gestures, and such high detection rate supports that fine-grained statistical information provided by FitCoach is reliable.

8.9. System latency

FitCoach aims to provide timely feedback to users. We define the system latency as the duration from the time point that users finish one repetition to the time point that our user interface shows the results. We randomly choose 30 repetitions from 6 volunteers and record the latency. We find that FitCoach achieves a latency of 0.8s on average. Therefore, we can conclude that FitCoach can provide timely result.

9. Discussion

Monitoring Non-arm Involved Exercises. Most wearable mobile devices, such as swartwatches and smartphones with armbands, are usually worn on human's wrists or upper arms, it thus becomes not easy to track/distinguish people's exercises that do not involve arms such as leg exercises (e.g., squat and front squat). However, with additional sensors attached on other parts of the human body, such as shoe sensors (Nike+ sensor) and waist/ankle smartbands (Flyfit), FitCoach can be easily extended to differentiate other non-arm involved exercises. By utilizing the same type of sensors (i.e., accelerometer and gyroscope), FitCoach can provide a more comprehensive picture to monitor and review users' fitness activities involving the whole body.

Energy Efficiency. Considering the limited battery capacity on wearable mobile devices, FitCoach has produced a set of solutions:

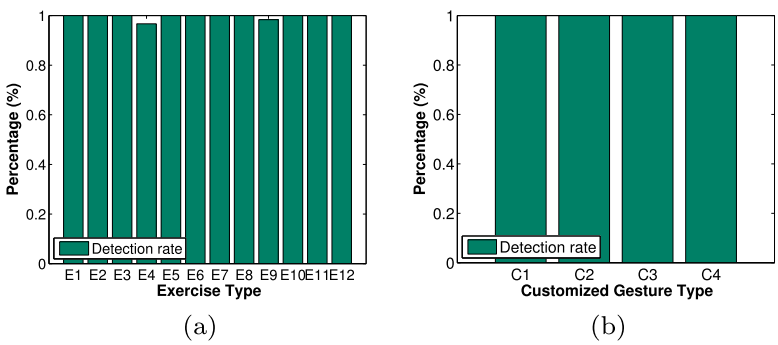


Fig. 18. Detection rate of exercise repetitions and customized gestures by using smartwatch.

First, FitCoach only cares about the workout data. The non-workout data is not processed after workout detection. Second, when distinguishing different types of exercises or customized gestures, the design philosophy in FitCoach is to use light-weight algorithms (e.g., light-weight SVM). Comparing to other techniques (e.g., Dynamic Time Wrapping (DTW)) being used in many existing gesture recognition works (Akl & Valaee, 2010; Liu, Zhong, Wickramasuriya, & Vasudevan, 2009), the proposed system doesn't involve much computational complexity. Third, people tend to perform the same type of reps within a set. To further reduce the computing complexity, FitCoach could use the first few rep segments to perform classification and determine the workout type based on the majority decision. Additionally, reducing the sampling rate of inertial sensors is another dimension that could be explored.

10. Conclusion

In this work, we propose FitCoach, an integrated mobile solution that can conduct systematic fitness monitoring and provide performance review based on a single off-the-shelf wearable device (e.g., wrist-worn wearables or arm-mounted smartphones). FitCoach has the capability to perform fine-grained exercise recognition including exercise types, the number of sets and repetitions by using inertial sensors from wearable devices without user involvement. Two novel metrics, exercise form score and workout review plane, are developed to provide effective review and recommendation for achieving effective workout and preventing injuries. It further enables contactless device control during workout through distinguishing customized gestures from regular exercise movements (e.g., arm swing during jogging). To ensure the system accuracy and robustness, FitCoach uses the earth/human coordinate system to align and integrate sensor readings from various device orientations. Extensive experiments involving 12 participants doing workout for over half a year time period demonstrate that FitCoach successfully takes one step forward to provide the integrated fitness monitoring system with over 90% workout analysis accuracy.

Acknowledgments

This work is supported in part by the NSF grants CNS1514436 and CNS1409767.

References

Abdelnasser, H., Youssef, M., & Harras, K. A. (2015). Wigest: A ubiquitous wifi-based gesture recognition system. In *Proceedings of the IEEE international conference on computer communications* (pp. 1472–1480). IEEE INFOCOM).

Akl, A., & Valaee, S. (2010). Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing. In *Proceedings of the IEEE international conference on acoustics speech and signal processing (ICASSP)* (pp. 2270–2273). IEEE.

Arney, J. (2005). You should be in pictures! experts share tips on how to produce a quality fitness video for any purpose. *IDEA Fitness Journal, 2*(1), 86–90.

Bouchard, C., Depres, J.-P., & Tremblay, A. (1993). Exercise and obesity. *Obesity, 1*(2), 133–147.

Chang, K., Chen, M. Y., & Canny, J. (2007). Tracking free-weight exercises. In *Proceedings of the 9th international conference on Ubiquitous computing*. ACM UbiComp).

Cheng, H.-T., Sun, F.-T., Griss, M., Davis, P., Li, J., & You, D. (2013). Nuactiv: Recognizing unseen new activities using semantic attribute-based learning. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM Mobisys).

Deller, J. R., Proakis, J. G., & Hansen, J. H. (1993). *Discrete time processing of speech signals* (1st ed.). Prentice Hall PTR.

D. P. Doane, L. E. Seward, Applied statistics in business and economics, USA: Irwin.

Doliotis, P., Stefan, A., McMurrough, C., Eckhard, D., & Athitsos, V. (2011). Comparing gesture recognition accuracy using color and depth information. In *Proceedings of the 4th international conference on PErvasive technologies related to assistive environments* (p. 20). ACM PETRA).

Fitbit. https://www.fitbit.com/.

Fitstar. http://fitstar.com/.

Flyfit. http://www.myflyfit.com/.

Hao, T., Xing, G., & Zhou, G. (2015). Runbuddy: A smartphone system for running rhythm monitoring. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*. ACM Ubicomp).

Jefit. https://www.jefit.com/.

Keally, M., Zhou, G., Xing, G., Wu, J., & Pyles, A. (2011). PBN: Towards practical activity recognition using smartphone-based body sensor networks. In *Proceedings of the 9th ACM conference on embedded networked sensor systems* (pp. 246–259). ACM SenSys).

Kurakin, A., Zhang, Z., & Liu, Z. (2012). A real time system for dynamic hand gesture recognition with a depth sensor. In *Proceedings of the 20th European signal processing conference* (pp. 1975–1979). IEEE EUSIPCO).

Ladabaum, U., Mannalithara, A., Myer, P. A., & Singh, G. (2014). Obesity, abdominal obesity, physical activity, and caloric intake in us adults: 1988 to 2010. *The American Journal of Medicine, 127*(8), 717–727.

Lindberg, B. (1971). On smoothing and extrapolation for the trapezoidal rule. *BIT Numerical Mathematics, 11*(1), 29–52.

Liu, J., Wang, Y., Chen, Y., Yang, J., Chen, X., & Cheng, J. (2015). Tracking vital signs during sleep leveraging off-the-shelf wifi. In *Proceedings of the 16th ACM international symposium on mobile ad hoc networking and computing* (pp. 267–276). ACM Mobihoc).

Liu, J., Zhong, L., Wickramasuriya, J., & Vasudevan, V. (2009). uwave: Accelerometer-based personalized gesture recognition and its applications. *Pervasive and Mobile Computing, 5*(6), 657–675.

Mifflin, M. D., St Jeor, S. T., Hill, L. A., Scott, B. J., Daugherty, S. A., & Koh, Y. (1990). A new predictive equation for resting energy expenditure in healthy individuals. *American Journal of Clinical Nutrition, 51*(2), 241–247.

Mokaya, F., Lucas, R., Noh, H. Y., & Zhang, P. (2015). Myovibe: Vibration based wearable muscle activation detection in high mobility exercises. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing* (pp. 27–38). ACM Ubicomp).

Nike+ sensor. https://secure-nikeplus.nike.com/plus/products/.

Parate, A., Chiu, M.-C., Chadowitz, C., Ganesan, D., & Kalogerakis, E. (2014). RisQ: Recognizing smoking gestures with inertial sensors on a wristband. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services* (pp. 149–161). ACM MobiSys).

Park, T., Lee, J., Hwang, I., Yoo, C., Nachman, L., Song, J., et al. (2011). A collaborative architecture for energy-efficient gesture recognition with hand-worn sensor and mobile devices. In *Proceedings of the 9th ACM conference on embedded networked sensor systems*. ACM Sensys).

Parthasarathy, S., Mehta, S., & Srinivasan, S. (2006). Robust periodicity detection algorithms. In *Proceedings of the 15th ACM international conference on Information and knowledge management* (pp. 874–875). ACM CIKM).

Pu, Q., Gupta, S., Gollakota, S., & Patel, S. (2013). Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM MobiCom).

Quaternion. https://en.wikipedia.org/wiki/Quaternion.

Ren, Y., Chen, Y., Chuah, M. C., & Yang, J. (2013a). Smartphone based user verification leveraging gait recognition for mobile healthcare systems. In *Proceedings of the 10th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks*. IEEE SECON).

Ren, Z., Yuan, J., Meng, J., & Zhang, Z. (2013b). Robust part-based hand gesture recognition using kinect sensor. *IEEE Transactions on Multimedia, 15*(5), 1110–1120.

Schölkopf, B., Sung, K.-K., Burges, C. J., Girosi, F., Niyogi, P., Poggio, T., et al. (1997). Comparing support vector machines with Gaussian kernels to radial basis function classifiers. *IEEE Transactions on Signal Processing, 45*(11), 2758–2765.

of Sports Medicine, A. C., et al. (2013). *ACSM's guidelines for exercise testing and prescription*. Lippincott Williams & Wilkins.

Suk, H.-I., Sin, B.-K., & Lee, S.-W. (2010). Hand gesture recognition based on dynamic bayesian network framework. *Pattern Recognition, 43*(9), 3059–3072.

Sundholm, M., Cheng, J., Zhou, B., Sethi, A., & Lukowicz, P. (2014). Smart-mat: Recognizing and counting gym exercises with low-cost resistive pressure sensing matrix. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing* (pp. 373–382). ACM Ubicomp).

Sworkit. http://sworkit.com/.

Thomaz, E., Essa, I., & Abowd, G. D. (2015). A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing* (pp. 1029–1040). ACM Ubicomp).

Vapnik, V. N., & Vapnik, V. (1998). *Statistical learning theory* (Vol. 1). New York: Wiley.

Vlasic, D., Adelsberger, R., Vannucci, G., Barnwell, J., Gross, M., Matusik, W., et al. (2007). Practical motion capture in everyday surroundings. In *, 26. ACM Transactions on Graphics* (p. 35).

Wang, Y., Liu, J., Chen, Y., Gruteser, M., Yang, J., & Liu, H. (2014). E-eyes: Device-free location-oriented activity identification using fine-grained wifi signatures. In *Proceedings of the 20th annual international conference on Mobile computing and networking* (pp. 617–628). ACM Mobicom).