2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)

Posture Tracking Meets Fitness Coaching: A Two-Phase Optimization Approach with Wearable Devices

Han Zhou*, Yi Gao*, Wenxin Liu*,Yuefang Jiang†, Wei Dong*
*Zhejiang University, China and Alibaba-Zhejiang University Joint Institute of Frontier Technologies, China
*Email: {zhouh,gaoy,liuwx,dongw}@emnets.org
†Soyea Technology, China, Email: jiangyf@soyea.com.cn

***Abstract*—Fitness training is becoming an increasingly popular way of maintaining overall health and preventing illness. However, in some cases the training could be risky and fitness-related injuries have increased by 48% in the USA. The training itself will not cause hurt, but if it is performed in a crucial-improper form (using the wrong technique), it will injure the exerciser. Research has explored the potential of using wearables to monitor fitness training, but the consideration of proper/improper form is not included. In this paper, we propose *WearCoach*, a wearable-based fitness training assistant, which acquires the user's fitness form information and generates real-time feedback during training. *WearCoach* differs from previous work of training assistant in 1) it employs a two-phase tracking algorithm to achieve accurate and real-time tracking of body motion, 2) it analyzes the captured arm posture and generates training guidance based on the user's form, 3) it uses joint orientation as an exercise classification feature to improve recognition accuracy. We conducted experiments with eight participants and nine exercises. Three kinds of feedback are generated, including injury alert, movement correction and symmetry analysis.**

***Index Terms*—Fitness Training Assistant, Motion Tracking, Multi Sensor Fusion, Wearable Computing**

I. INTRODUCTION

Extensive studies have investigated how physical activity helps people enhance overall health and prevent chronic illness [12]. For example, regular fitness training lowers the risk of cardiovascular disease [13] and pulmonary disease [7]. With the development of the fitness industry, the popularity of exercise is increasing through the provision of fitness facilities. In 2014 over 60% of Americans aged 6+ years took part in fitness exercise, which is regarded as the most popular physical activity [3]. Among the great diversity of exercises, most of them require the exerciser to perform certain movements with proper form. If the exercise is not performed with proper form, the training will be significantly less effective or even cause risk of injuries, especially during weight training [20]. Figure 1 shows the comparison of dumbbell lateral raise performed in a proper and an improper form. When the exerciser raises the elbow much above the shoulder, it causes a high shoulder injury risk [16].

In the past decades, researchers and start-ups have studied how to provide exercisers with training assistance and injury prevention. For example, a Kinect application "Your Shape: Fitness Evolved" uses a Kinect to capture the user's body motion. The tracked posture is displayed beside the coach's posture that the user will notice the improper form as shown in Figure 2. However, computer-vision-based approaches require the deployment of cameras, which draws the concern of cost and privacy, hence are not suitable for some public places, e.g., a gym. Benefited by the increasing computing ability, attached rich sensors and the close to human body position, wearable devices (or *wearables* for short) show a unique potential for fitness training monitoring. In reality, tracking body motion using wearables has been well investigated [5], [19], [23]. Since these studies do not target training monitoring, they are subject to different limitations. RF-Kinect [23] and wearable body sensor networks (BSN) [5] place one wearable for each main joint, e.g., the elbow and wrist, such implementation would introduce more inconvenience and can't be applied in the most popular commercial wearables like a smartwatch or smart bracelet. ArmTrak [19] uses a commercial smartwatch to track the arm posture, which could be leveraged for evaluating the exercise. However, the tracking accuracy of ArmTrak is limited -median error above 13cm or 30 degrees in the real-time tracking scenario, and this may lead to a misguidance.

Fig. 1: Comparison of dumbbell lateral raise performed in proper and improper form.

Fig. 2: The display of "Your Shape: Fitness Evolved" where the user can observe the improper form.

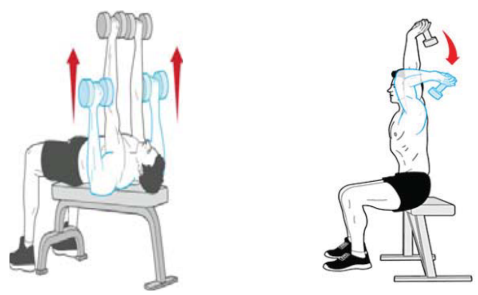
In this paper, we propose WearCoach, a fitness training assistant implemented by two wearables, which targets tracking and analyzing the user's arm motion and providing real-time guidance for the user to improve training effect and prevent injury. Through analyzing the movement pattern of

2155-6814/20/$31.00 ©2020 IEEE
DOI 10.1109/MASS50613.2020.00070

different exercises, WearCocah proposes using the feature *joint orientation* for exercise classification. The joint orientation provides more information compared with accelerometer or gravity sensor readings, and it is capable of recognizing the exercises with higher precision. In order to cope with the limited sensor measurements, WearCoach presents a two-phase tracking algorithm to estimate the arm posture. In phase one, we constrain the elbow position into a small space with knowledge of the exercise, which efficiently reduces the search space and further computation. In phase two, a gradient descent algorithm is used to calculate the optimal result based on the distance measurement between the two wrists. This two-phase algorithm transforms the tracking of the two arms into one problem and significantly reduces the computation overhead.

We summarize the contributions of our work as follows:

- We propose a two-phase algorithm for arm postures tracking using wearables. Through combining the tracking of each arm into an optimization problem, the tracking results show a high degree of accuracy.
- We propose to use the feature joint orientation to improve the exercise classification performance. That is because the unique movement pattern of different exercises will lead to different joint orientation changes, and compared with a three-dimensional inertial sensor reading the joint orientation contains more information about the movement.
- We implement a prototype of WearCoach, which records precise fitness statistics and generate well-understood guidance including injury alert and movement correction.



(a) (b)

Fig. 3: Illustration of (a) Dumbbell bench press; (b)Tricep press (image source [11]).

The rest of the paper is organized as follows: In Section II, we first give a short introduction to fitness training, then we analyze the joint orientation change during different exercises. In Section III, we present the WearCoach system architecture. Then we explain the recognition process and the tracking process in Section 4 and Section 5, respectively. The evaluation of WearCoach is in Section 6. In Section 7, we introduce the related work. In the end, we conclude this paper in Section 8.

Starting position

1.Dumbbells should be rotated so that both handles are in line with one another; simulate a barbell with its center bar running through both dumbbell handles.

Upward Movement

2.Your wrists should remain firm and straight, forearms almost perpendicular to the floor and the hands aligned with each other.

3.Press the dumbbells upward until the elbows are fully extended but not locked.

Downward Movement

4.To maintain a stable position on the bench, lower both dumbbells at the same rate.

5.Lower the dumbbells to a lateral portion of the chest near the armpit, even with nipple level.

Fig. 4: Guidance of dumbbell bench press to maintain proper form.

II. PRELIMINARY

In this section, we introduce basics of exercise-related injury prevention and movement pattern of exercises.

A. *Exercise-Related Injury Prevention*

In fitness training, there is always a potential of injury and that may lead to disturbance to daily life, loss of wealth, possible disability, or in severe cases, death. According to the epidemiology study [9] that 36% of the exercise-related injury is caused by overexertion or unnatural movement, which is much higher than the secondary injury cause.

In order to have an understanding of form, we use dumbbell bench press to express what requirements should be satisfied to maintain proper form in a particular exercise. The guidance of the dumbbell bench press is listed in Figure 4 refer to professionals [8]. From the guidance, it can be observed that proper form requires the control of muscles and let the body move in a fixed trajectory, the symmetry of the body and other factors like breath pattern. Without consideration of the breath pattern, the evaluation of exerciser's form can be regarded as the evaluation of the body move trajectory which can be represented by joint angles.

B. *Movement Pattern*

One target in fitness training monitoring is to know what exercise the user is doing. Previous works have explored extensive features based on accelerometer readings [2], [10] for exercise classification. MiLift [18] extracts features from gravity sensor readings because they reflect the wrist orientation. However, these sensor readings only partially represent the body trajectory and cannot work for some fitness movements. First, we clarify the definition of joint orientation. Figure 5 (a) shows we establish a wrist coordinate system (WCS) based on the wrist and use this WCS to represent the wrist orientation Ori_W. The WCS sets the wrist as the origin, the positive X axis along the direction from the elbow to the wrist, the positive Z axis be the line which emanates from the wrist and is perpendicular to the wrist plane, and the Y axis be the line perpendicular to the XZ plane. Here for simplicity, we let the user wear a smartwatch on the left wrist, then the WCS is the same as the device coordinate system (DCS). Similarly, we can define other joint orientations like the elbow or knee orientation.

More specifically, we define the user torso coordinate system (TCS) as the reference coordinate system in Figure 5 (b). This TCS sets the user left shoulder as the origin, the line emanating from the shoulder in the leftward direction as the positive X axes, the line emanating from the shoulder in the frontward direction as the positive Z axes and the line perpendicular to the XZ plane in the upward direction as the positive Y axis. Then we use vectors $\vec{X}$,$\vec{Y}$ and $\vec{Z}$ in the TCS to represent the X, Y and Z axes of the wrist coordinate system. A unique tuple ¡$\vec{X}$,$\vec{Y}$ and $\vec{Z}$¿ is regarded as a unique wrist orientation Ori_W and we use $Ori_W(X)$ to represent the $\vec{X}$ of the tuple.

(a) (b)

Fig. 5: (a) Wrist coordinate system (device coordinate system); (b) User torso coordinate system.

Reviewing the descriptions of dumbbell bench press, we find that particular exercise follows a fixed movement pattern. Figure 3 displays two movements dumbbell bench press and tricep press. We observed that the joint orientations (wrist orientation and elbow orientation) of two movements show the particular pattern. In dumbbell bench press, the elbow orientation changes with the increasing of shoulder extension angle (around 90 degrees), and the wrist orientation barely changes. In tricep press, the wrist orientation changes with the increasing of elbow flexion angle (around 90 degrees), and the elbow orientation does not change. It implies 1) only limited joint orientations appear during one movement, 2) joint orientations of different movements have significant differences. Therefore, we could identify different fitness movements by analyzing joint orientations.

III. System Overview

WearCoach focuses on accurately recording the body trajectory during fitness training, and through analyzing the postures it provides useful guidance for the user to improve exercise and keep the potential injury risk away.

Figure 6 shows the architecture of WearCoach. The user is asked to wear two wearable devices on each wrist while exercising. The wearables have a built-in IMU, a microphone and a speaker, and the IMU outputs are sent to WearCoach continuously. In Fitness Training Detection&Segmentation process WearCoach analyzes the pattern of IMU data and tries to capture the training state. Once the training state is detected, the acoustic ranging process is awakened, and WearCoach collects the distance measurement between two wearables. The distance measurement is analyzed with IMU readings to detect the start and the end of each repetition accurately. The Fitness Training Classification process extracts wrist orientations as a feature from IMU readings and uses a support vector machine (SVM) classifier to identify the exercise type. Besides, it recognizes the exercise mode by analyzing the difference between IMU readings from two wearables. A two-phase tracking algorithm is implemented to track arm postures with limited measurements. After recognizing the exercise type, the prior knowledge of the exercise is used to reducing the search space for tracking. Then we transform the tracking of the two arms into an optimization problem to fast the calculation. The feedback contains fitness statistics and guidance. The guidance is generated by analyzing the continuous arm postures referred to medical criterion and expert knowledge of proper form. Without loss of generality, we only discuss the usage of WearCoach for evaluating arm-related fitness movements, which are typical and representative. Via the techniques in this paper, WearCoach can be used on more fitness movements if the wearables are attached to relevant parts of the body.

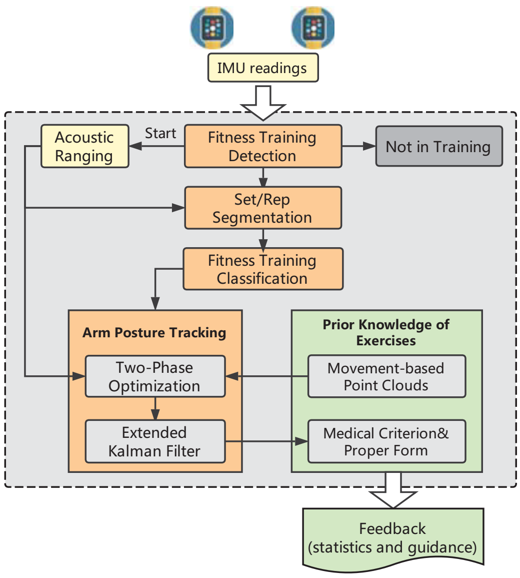


Fig. 6: Architecture of WearCoach.

IV. Joint Orientation-based Exercise Recognition

Some important notations used in Section IV and Section V are listed as follows. In this paper, we use subscript 'l' and 'r' to represent the left and right arm, 'e' and 'w' represent the elbow and the wrist.

- $\boldsymbol{Quat}$: Quaternion to represent joint orientation, which is composed of a real number part q_0 and a vector part $\mathbf{q} = [q_1, q_2, q_3]$.

- $\boldsymbol{Z}$: Position of the joint, which is a 3D vector $\boldsymbol{Z} = [Z^x, Z^y, Z^z]$
- $\boldsymbol{LA}$: Linear acceleration of the joint, which is a 3D vector $\boldsymbol{LA} = [LA^x, LA^y, LA^z]$.
- $\boldsymbol{\omega}$: Gyrometer outputs in three axes, $\boldsymbol{\omega} = [\omega^x, \omega^y, \omega^z]$.
- $\boldsymbol{A}$: Accelerometer outputs in three axes, $\boldsymbol{A} = [A^x, A^y, A^z]$.
- $\boldsymbol{D}$: Acoustic ranging module outputs which implies the distance between two wearables.
- $\boldsymbol{L_{upper}}$: Upperarm length of the user.
- $\boldsymbol{L_{fore}}$: Forearm length of the user.

A. Fitness Training Detection&Segmentation

1) Detection: Users would prefer the exercise assistant that can work automatically, without manually input or select. So the first task for WearCoach is to automatically detect fitness training, which has been discussed in the literature. For example, FitCoach [10] detects the long-term periodic features in accelerometer readings to judge the appearance of exercise, MiLift [18] uses Conditional Random Fields (CRF) and other classifiers like Decision Tree (DT) to label a 1-second window of accelerometer data. We borrowed the technique from FitCoach to detect fitness training. Once the process detects fitness training, WearCoach starts sending the IMU readings to the Segmentation process and turns on the ultrasonic distance measuring module.

2) Segmentation: People tend to segment fitness training exercise in two levels, set and repetition. A repetition/rep means a complete instance of the fitness training movement. A set means an exercise session which includes a bunch of repetitions. It is easier to identify two sets and the rest session between them. The results of Exercise Detection process would deliver if it is in the fitness training state in each time slot, and we can add continuous fitness training time slots to form the fitness training set.

However, accurately detecting each repetition during fitness training seems not straightforward. Existing methods try to analyze the pattern of accelerometer readings [2] or calculate the correlation between continuous sensor data samples. They meet the goal of counting the number of repetitions, but they can not identify the exact start and end time of each repetition. Analysis of the original accelerometer reading may work, but it is also sensitive to the noise. Since WearCoach collects the distance between wrists for arm posture tracking, we use a double-check method to achieve accurate rep segmentation. The intuition is that at the start and the end of the movement, user's wrists are at the same position which is called the start position (initial position). By detecting the IMU reading change, we can find the start time point of the first repetition, and then we record the wrist distance of the start position. For the following repetitions, we judge the end of the repetition by 1) detecting no accelerometer readings but gravitational acceleration, 2) current wrist distance is close to the recorded one.

B. Fitness Training Recognition

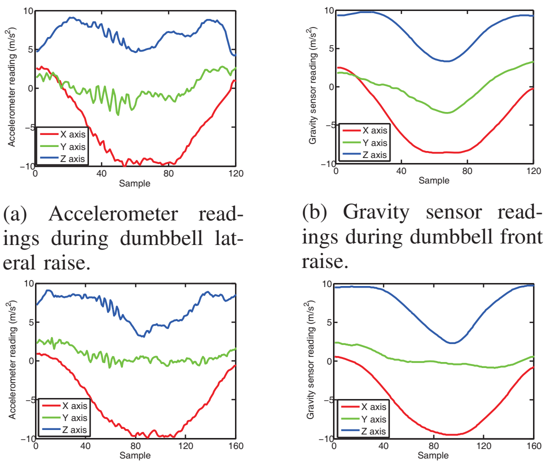
As mentioned in II-B, WearCoach uses joint orientations to distinguish different fitness movements. Specifically, we use wrist orientation as the feature because it can be directly accessed from the IMU. We have explained why joint orientation is a useful feature when doing such a classification. Here we compare it with other features in the literature and show the difference. Three kinds of features are discussed, including the accelerometer reading-based features, gravity sensor reading-based features and joint orientation-based features.

First, we take a deeper look at the origin data and find what it means during fitness training. For example, we assume a user is exercising with a smartwatch on the left wrist. The 3-axis gravity sensor readings are determined by the local gravitational acceleration and the wrist orientation. The algorithm will first estimate the device/wrist orientation and then multiply the estimated orientation vectors and the gravitational vector, respectively, which derives the software-defined gravity sensor readings. The 3-axis accelerometer readings are determined by the linear acceleration caused by the user, the local gravitational acceleration and the wrist orientation. It contains complex situations because when the user pauses at the start or end position for some rest, the readings only include gravity sensor readings, which are determined by the wrist orientation. Further, when the arm is moving the accelerometer readings reflect the user's power to move the weights. So how the user moves the weights, fast or slowly, will directly affect the 3-axis accelerometer readings. The gravity sensor readings and the accelerometer readings are both determined by the wrist orientation, but they can't represent the complete wrist orientation. That is, when the user lets her/his arm free-fall on the left side of the body, X axis readings of the accelerometer and the gravity sensor are the same, i.e. local gravitational acceleration, and the Y and Z axis readings are 0. What we can conclude is only that X axis is pointing at the gravitational direction, but we don not know which direction the other two axes are pointing at.

Figure 7 displays the accelerometer readings and gravity sensor readings change during two kinds of fitness movements, dumbbell lateral raise and dumbbell front raise. They share a similar movement pattern that is lifting the wrist from free-fall to the same height as the shoulder while keeping the arm straight. The difference is that for dumbbell front raise the arm points a front direction at the highest position, but the dumbbell lateral raise points a left direction. Figure 7 (a) and (c), (b) and (d) show that it is hard to distinguish these two exercises because the sensor readings are so close. However, it becomes distinguishable when using wrist orientation since they are pointing at different directions.

V. Two-Phase Posture Tracking

In this section, we introduce the posture tracking algorithm. First, we clarify the definition of an arm posture. An arm posture means a static 3D model of the arm, which is uniquely defined by an elbow and wrist orientation pair. When the user's forearm and upper arm length are available, the posture



(a) Accelerometer readings during dumbbell lateral raise.

(b) Gravity sensor readings during dumbbell front raise.

(c) Accelerometer readings during dumbbell lateral raise.

(d) Gravity sensor readings during dumbbell front raise.

Fig. 7: Sensor reading changes during dumbbell lateral raise and dumbbell front raise.

can also be used to derive the relative positions among the shoulder, elbow and wrist.

A. *Movement-based Point Clouds*

ArmTrak [19] proposes point clouds, which are useful tools in arm posture tracking. The theoretical basis for point clouds is 1) human joint rotation is limited by the range of motion (RoM) [6], 2) wrist orientation is determined by the combination of the shoulder rotation and the elbow rotation. This indicates that for a wrist orientation there existing limited arm postures, i.e., limited elbow orientations. A point cloud means a wrist orientation and the possible elbow orientations corresponding to it. Then the set of every point cloud is called point clouds.

However, the accuracy and latency trade-off exists when using point clouds. Point clouds with finer granularity lead to a better accuracy but much higher latency and more storage requirement. Our target is to achieve real-time and high precision posture tracking with mobile devices, so the computation resources and storage are limited.

Remind the feature we mentioned in Section 2, each fitness training exercise follow its specific movement pattern. Therefore, we propose producing movement-based point clouds to assistant arm posture tracking in fitness training. Before presenting the details of movement-based point clouds, one assumption should be stated that the user has a basic understanding of the exercise and will not perform it in a way too different from the standard movement. Because WearCoach tries to help experienced exercisers to achieve self-guided improving, when a user has none basic concept of fitness training, it is very dangerous to let her/him exercise alone.

Now we introduce how to establish movement-based point clouds. First, we build the whole point clouds like ArmTrak [19] did, which contain all possible arm postures. Then for a certain fitness training movement, we build its joint orientation description, which is a set that contains possible wrist and elbow orientation pairs when performing this movement. The joint orientation description is built as follows: 1) we ask an expert trainer to perform the movement with wearables attached on the wrist and elbow of each arm; 2) the wearables continuously calculate and record the instant joint orientations; 3) we collect the joint orientation results and manually add biases to them. This joint description leverages our prior knowledge of the movement pattern and it contains the most possibly appeared arm postures during the movement. In order to handle the movement in improper form performed by the user and the physical differences between different users, we enlarge the joint description by adding biases to the orientations. In our implementation, the bias is set below 20 degrees in each axis. Notice the adding bias process will produce some orientation pairs out of the joint RoM, and we need to remove them. For each wrist and elbow orientation pair in the description set, we map it to an orientation pair in the point clouds. The mapping is based on calculating the distance in orientation space, and we choose the orientation pair with the smallest distance. The mapping procedure will help us label the orientation pairs in the point clouds. For example, once an item in the point clouds is mapped to the joint orientation description of exercise A, we could label this item with A. In this paper, we investigate nine representative upper body fitness exercises which are dumbbell biceps curl, dumbbell bent-over row, dumbbell bench press, lying triceps extension, dumbbell front raise, dumbbell lateral raise, dumbbell shoulder press, overhead triceps extension, cable cross-over.

Considering the great diversity of fitness training exercises, we do not add a label column for each exercise for storage concern. For simplicity we add a column 'type' in the point clouds and use an int value to represent the label. Each bit of the int would represent an exercise. When the bit is set 1, it means this item is labeled with this exercise and 0 when not labeled. The obvious benefit of using movement-based point clouds is that it achieves both high accuracy and low latency. Original point clouds with fine granularity will return too many candidate arm postures for a given wrist orientation, which requires too much time to run a tracking algorithm like a particle filter. The exercise label first assists speeding up the query, and the returned candidates are much less, so they allow us to run a tracking algorithm to calculate the optimal estimation.

B. *Tracking solution*

Here we would like to clarify the method that determines the arm posture based on limited measurements. For simplicity, we assume the user is doing exercise in single mode with the left arm. After the segmentation and recognition process, we assume the right exercise type is recognized. So once a wrist orientation is ready, we query the movement-based point clouds as mentioned in V-A, and a set of candidates is returned.

A simple method is choosing the candidate that mostly matches our measurements. However, this method presents weaknesses in accuracy and latency. Theoretically, for an acquired wrist orientation, we should return the candidates with the same wrist orientation. In practice, the point clouds are formed in advance with certain granularity. For storage reason, the point clouds granularity is usually 5 degrees or larger. It usually happens that there are no such candidates and we need to look for the nearest neighbors. As for the computation, we need to compute for every candidate and it costs more time when more candidates are returned. So how should we utilize the point clouds or the candidates? The intuition is that even the exact arm posture may not be in the candidates, it must be surrounded by them. Remind that wrist and elbow orientation can completely express an arm posture, if we regard the wrist orientation is error-free, then the problem turns into finding the right elbow orientation. In practice, the wrist orientation contains error due to IMU noises and improper placement. However, with advanced orientation estimator [25] we can acquire accurate orientation with an angle error less than three degrees. Therefore, in this paper we assume we get the accurate wrist orientation and based on this we estimate the elbow orientation. It is natural to infer that the exact elbow orientation is similar to the estimated elbow orientation candidates. Thus if we use elbow orientations of the candidates to form an orientation space, the true elbow orientation is within this space. We propose a two-phase tracking solution to solve this problem.

Phase 1: In this phase, we build an elbow position space based on the candidates. As mentioned above, the elbow orientation of candidates could form an orientation space. For ease to calculate, we transform this elbow orientation space into the elbow position space by multiplying the X axis vector and the upper arm length. We indicate that the exact elbow position is within this space.

Phase 2: In this phase, we use a gradient descent method to calculate the optimal elbow position. The optimal elbow position estimation Z_{le} and Z_{re} must conform to our measurements including the distance measurement D and the acceleration measurement A_w. The distance measurement D represents the distance between two wearables, e.g. between two wrists and can be formulated by

$$\begin{aligned} D &= \left| \, Z_{lw} \; - \; Z_{rw} \, \right|_2 \\ &= \left| \, Z_{le} + Ori_{lw}(X) \cdot L_{fore} - Z_{re} - Ori_{rw}(X) \cdot L_{fore} \, \right|_2. \end{aligned} \tag{1}$$

The acceleration measurement A_w consists the linear acceleration of the wrist LA_w and the gravitational acceleration G as

$$A_w = LA_w + G \cdot R(TCS,\ WCS), \tag{2}$$

where $R(TCS,\ WCS)$ represents the rotation matrix from the torso coordinate system to the wrist coordinate system. For every elbow position estimation pair $< Z_{le}^{est}, Z_{re}^{est} >$, the corresponding D^{est} and A_w^{est} could be calculated by Equation 1 and 2. The lost function is based on the difference between the measurements and the corresponding estimations derived by the elbow position estimations. We generate different initial elbow position estimations to make the algorithm find the global optimal solution.

$$\begin{aligned} Lost =& ((D^{est} - D)/D)^2 \cdot ((A_{lw}^{est} - A_{lw})/A_{lw})^2 \\ & \cdot ((A_{rw}^{est} - A_{rw}) \, / \, A_{rw})^2. \end{aligned} \tag{3}$$

Fig. 8: WearCoach prototype. A ultrasonic distance measuring module KS102 and an IMU sensor MPU-9250 are used.

VI. IMPLEMENTATION & EVALUATION

A. Implementation

Hardware. In Figure 8, we display the prototype of WearCocah. We implemented WearCoach with the development board STM32L151. An IMU sensor MPU-9250 was attached to each board by soldering, which consists of a 3-axis accelerometer and a 3-axis gyroscope. WearCoach adopts an ultrasonic distance measuring module to achieve acoustic ranging. This module has a specialized speaker which emits 40KHz ultrasonic signals and a specialized microphone which is able to capture these signals. Experiment results show that distance measuring error in WearCoach is 1cm on average.

Software We developed the fitness training recognition and tracking program on the smartphone in Android and C languages. Especially, in fitness recognition, we used a Support Vector Machine (SVM) classifier to determine the exercise type. The input of the SVM are $Ori(X)$ and $Ori(Y)$ of the wrist orientation (two axes could uniquely identify a three-dimension Cartesian coordinate system) and the kernel function is a radial basis function. We also trained another two SVMs which take the accelerometer or gravity sensor readings as input to do recognition for comparison. Otherwise, we did modifications to the real-time version ArmTrak [19] by replacing the original point clouds with movement-based point clouds.

Methodology We asked eight volunteers to take part in our experiments, including six males and two females. Before experiments, we tested them by a questionnaire to make sure they have no physical injury and are capable of doing normal fitness training movements. They are asked to wear the WearCoach prototype on each wrist and did the nine exercise mentioned in Section V-A. We divided the experiments into two parts. In the first part, the participants exercised with an expert trainer, and the trainer helped them to improve the movements. At the end of this part, the expert trainer guided each participant to finish the movement in proper form, and WearCoach recorded the movement data as the individual standard movement.

In the second part, we asked the participants to do the same exercises alone while WearCoach keeps collecting the fitness statistics. Every participant is required to do each exercise for three sets and each set contains eight repetitions, which is over 1700 reps for nine exercises. We manually labeled the exercise type for each repetition to acquire the classification groundtruth. We put a Kinect V2 in front of the participants, which keeps recording the 3D posture of the whole body. The Kinect data of arm posture is used as groundtruth. After the experiments, we fed the recorded sensor data to the modified Armtrak [19] and the SVMs adopting accelerometer/gravity sensor readings to gain tracking and classification results.

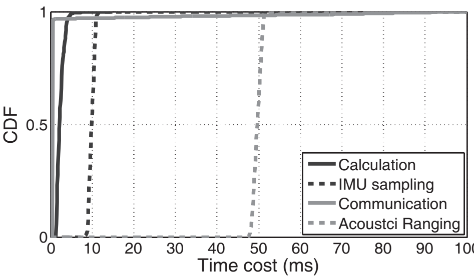


Fig. 9: Each part of time cost in WearCoach.

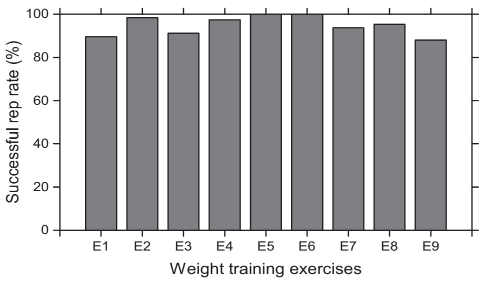
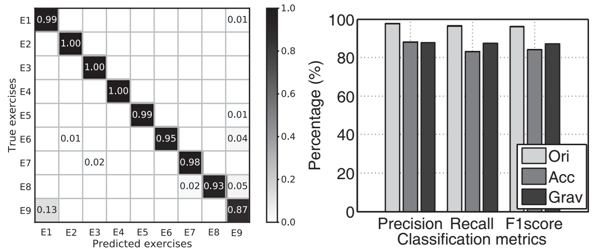


Fig. 10: The successful rep rate among nine exercises.

B. System Latency

We evaluate the system latency of WearCoach, which is an important metric in HCI applications. In order to achieve real-time coaching, WearCoach needs to track the arm posture in a high frequency and analyze it in the meantime. We define the system latency as the total time is required to produce a new posture estimation and finish the analysis mentioned in VI-E. In Figure 9, we plot the time-related results in experiments. The sensor data are updated every 10ms and 50ms for the IMU and the ranging module. Compared with them, the calculation time cost is below 5ms in all experiments for once posture tracking and analyzing. The communication time cost varies from 1ms to 80ms since it may be affected by environmental factors like wireless interference. However, the bad cases are rare, and at most of the time it only requires 1ms to send origin data from the wearable to the phone.

The conclusion is, the latency of WearCoach is not determined by the calculation time cost. WearCoach could support a 100Hz posture updating frequency if needed. Nevertheless, the arm moving speed is constrained by the weight in practice. We assume the moving speed is lower than 1m/s, which means a small movement below 1 cm when postures are updated at 100Hz. Since such granularity is too precise, we choose 20Hz as the updating frequency of WearCoach, which equals the acoustic ranging updating frequency. With this setting, the movement distance of two successive arm postures is below 5cm, which is precise enough for our analysis.



(a) (b)

Fig. 11: (a) Confusion matrix of classification results in WearCoach; (b) Average precision, recall and F1 score using three kinds of features over nine exercises.

C. Movement-based Point Clouds Evaluation

Here we want to prove our assumption in V-A that a user with basic knowledge of fitness training will perform the movement in our movement-based point clouds. We define the occasion that all arm postures during exercise are within the corresponding point clouds as a successful repetition for movement-based point clouds. Figure 10 displays the results of successful reps among nine exercises. For E5 and E6 it presents a 100% successful reps. For E9, which has the most failed reps, shows an 88% successful reps. The average successful rep rate is 94%. After analyzing the failed reps, we suggest it is related to the user's experience. For participants without much fitness training experience, they usually could not follow the movement pattern for the whole training duration. For example, one participant usually performed an unnecessary wrist rotation during E1 and this failed most of his reps. Since the movement-based point clouds generally work well for novices with once guiding in advance, it will achieve a better success rate when they are more experienced.

D. Classification Evaluation

Figure 11 (a) shows the classification result for WearCoach. The average precision is 96.7% among nine exercises. For E1 to E7, the precision is above 95%, which means WearCoach could efficiently capture these exercises. The worst case is of E9, where 13% reps was regarded as E1. E1 and E9 have a common movement pattern that they both rotate the elbow and lift the wrist, but the difference is that during E1 the $Ori_W(X)$ generally does not change while E9 does. When the user does not rotate the shoulder as expected in E9, it would be very similar to E1.

Figure 11 (b) displays the average precision, recall and F_1 score when using joint orientation, gravity sensor readings and accelerometer readings-based features for classification. The three metrics of the joint orientation-based feature are all above 95%. The precision for the accelerometer reading-based features is 88%, which is higher than the gravity sensor reading-based features. However, the recall and the F_1 score are lower which are only 83% and 84%. For the exercise we mentioned in IV-B, we find that a 20% and 26% mismatches between E5 and E6 for accelerometer reading and gravity sensor reading-based features. Using the joint orientation-based feature, they were perfectly recognized. Therefore, we conclude that joint orientation is a useful feature in exercise classification.

E. Case Study

Here we study what kind of coaching function could WearCoach provide and how does it help the user achieve better training.

Injury alert. The injury alert will be announced when WearCoach detects that the movement user is performing results in high injury risk. The feedback is valuable since it helps the user avoid the potential physical injury. Thus we should concern the alert speed which needs to be as fast as possible to prevent a sudden injury. The form of the alert also matters. We have different mobile devices in the scene and many interaction approaches, but which one should we choose?

WearCoach detects the injury-causing movements by analyzing the real-time arm postures. We collect the expert knowledge and medical criterion of the movements and turn these rules into a joint orientation description of the injury-causing arm postures. The analysis is easy to be done by calculating if the tracked arm posture conforms to these descriptions. We list some of the rules:

1. Rotation angles of shoulders and elbows exceed the RoM [6].

2. Moving the weight in front instead of up in dumbbell shoulder press [22].

3. Elbows are flaring in dumbbell bench press [15].

The smartphone display, voice reminding and vibrating of the wearables are all available approaches in our implementation. We choose the voice reminding considering the safety and speed need. The vibrating warning of the wearables makes the user notice at the soonest, but when the user is focusing on weightlifting the vibration may harm. It would cause an injury if the user loses balance because of the vibration. Moreover, the user may not look at a display all the time during training, so the display warning has drawbacks in speed. We choose the voice reminding as the injury alert approach which satisfies both requirements about safety and efficiency.

We asked the participants to perform injury-causing movements without any weights. For each fitness movement, 40 reps are performed by four participants. The results show that 100% of the movements caused an injury alarm, which shows the tracking accuracy of WearCoach satisfies the requirement of detecting such injury-related occasions. Benefited by the injury-related posture description, the injury alert process costs less than 1ms on average.

Movement correction. By movement correction, the user would notice the difference between the performed movement and the standard movement. Then, the user could improve the movement by correcting the different part.

First, we need to collect the standard movement data for each exercise and each user. We take the physical differences of individuals into consideration, which means the standard movement for others may not be your best one. Therefore, we recommend the user to upload the standard movements for herself/himself. Same as the first part in our experiments, the user is expected to exercise with a real coach beside, while WearCoach keeps recording the movements. We let the coach judge the standard repetition which is in proper form and label it. Next time when the user does self-training, WearCoach will display the standard movement and the tracked movement. Two 3D human models will be leveraged for displaying, and the arm movement will be demonstrated clearly.

Four participants were asked to do the same exercise while looking at the display of movement correction. For each exercise, 30 reps are performed as in the second part of the experiments. We compared the success rate of the movements performed with and without movement correction, find that 3% increase for the movements with correction. It proves that the movement correction feedback is useful for the user to improve the movement.

Moreover, WearCoach provides a symmetry analysis of the user's movement form. The balance factor is important in exercise because the imbalanced muscle power and body movement may weaken the exercise performance. WearCoach compares the postures between the two arms when training in the symmetry mode. Then the differences of rotation of joints, moving speed and range of motion could be calculated, which is useful for a user to optimize the fitness training plan and rebalance the movement and power.

VII. Related Work

Exercise assistant: [2] used a mobile phone and its built-in accelerometer to count the repetitions during exercise, but the peak detection method cannot identify the exact time of the start and end of the move. MiLift [18] proposed a two-level classification framework for exercise recognition, and it used gravity sensor readings which is useful in classifying weightlifting exercises with wrist movements.FitCoach [10] proposed a new metric exercise form score, which evaluates the workout by comparing motion strength and performing period of each repetition with a baseline. Therefore, existing wearable-based fitness training assistants support recording training data such as exercise type, exercise time, sets and reps. These training data is useful for fitness coaching because exerciser could evaluate the training program and prevent overtraining. However, these works do not support evaluating the user's form during training. Oli [21] acquired visual human skeleton by Kinect and performed a frame by frame analysis to generate metrics and injury alerts. The Kinect has applications and games for fitness in the marketplace, like Your Shape: Fitness Evolved and Nike+ Kinect Training, where a virtual trainer performs the standard move in a display and exercises with the user.

Wearable-based activity recognition: Most existing works target recognizing daily activities including walking, sitting, reading and so on. Dong *et al.* [4] recognizes the eating gesture and detects an individual's caloric intake via a wristwatch. BodyScope [24] uses a wearable acoustic sensor to record the sounds produced in the user's throat area and classify them. Attal *et al.* [1] recognize multiple physical activities with wearable sensors, but it requires multi wearable sensors placed on the user's body. COSAR [17] is a framework that provides context-aware activity recognition using statistical and ontological inference in the Android platform. PACT [14] presents how to assess smoking using automatic recognition of smoke inhalations in a smartwatch. Existing works show a promising ability to use wearables to recognize high-level human activities. Somehow they usually use the original IMU sensor readings to do such classification. If we walk one more step and try to distinguish the activities in finer granularity, joint orientation-based features may be more helpful.

VIII. CONCLUSION

In this paper, we propose WearCoach, a fitness training assistant implemented by wearable devices. WearCoach could automatically execute fitness training detection, segmentation, classification, tracking and analyzing, after which it records fine-grained fitness statistics and generates useful and well-understood guidance for the user. Based on the inertial and acoustic sensing measurements, WearCoach tracks the body trajectory. A two-phase tracking algorithm was proposed to handle the large search space of motions. In experiments, the average classification precision, recall and F_1 score are all above 95% among nine exercises. WearCoach provides different kinds of guidance based on analysis results. The injury alert will be helpful to detect and injury-causing movement and warn the user in time. The movement correction will help the user achieve movement improvement by showing the standard movement and the tracked.

ACKNOWLEDGMENT

This work is supported by the National Key R&D Program of China under Grant No. 2019YFB1600700 and National Science Foundation of China (No. 61872437).

REFERENCES

[1] Ferhat Attal, Samer Mohammed, Mariam Dedabrishvili, Faicel Chamroukhi, Latifa Oukhellou, and Yacine Amirat. Physical human activity recognition using wearable sensors. *Sensors*, 15(12):31314–31338, 2015.

[2] Keng-Hao Chang, Mike Y Chen, and John Canny. Tracking free-weight exercises. In *International Conference on Ubiquitous Computing*, pages 19–37. Springer, 2007.

[3] Physical Activity Council. Participation report, 2014.

[4] Yujie Dong, Adam Hoover, Jenna Scisco, and Eric Muth. A new method for measuring meal intake in humans via automated wrist motion tracking. *Applied psychophysiology and biofeedback*, 37(3):205–215, 2012.

[5] Alessandro Filippeschi, Norbert Schmitz, Markus Miezal, Gabriele Bleser, Emanuele Ruffaldi, and Didier Stricker. Survey of motion tracking methods based on inertial sensors: a focus on upper limb human motion. *Sensors*, 17(6):1257, 2017.

[6] Richard L Gajdosik and Richard W Bohannon. Clinical measurement of range of motion: review of goniometry emphasizing reliability and validity. *Physical therapy*, 67(12):1867–1872, 1987.

[7] Judith Garcia-Aymerich, Peter Lange, Marta Benet, Peter Schnohr, and Josep M Antó. Regular physical activity modifies smoking-related lung function decline and reduces risk of chronic obstructive pulmonary disease: a population-based cohort study. *American journal of respiratory and critical care medicine*, 175(5):458–463, 2007.

[8] John F Graham. Dumbbell bench press. *Strength & Conditioning Journal*, 22(4):71, 2000.

[9] Shannon E Gray and Caroline F Finch. The causes of injuries sustained at fitness facilities presenting to victorian emergency departments-identifying the main culprits. *Injury epidemiology*, 2(1):6, 2015.

[10] Xiaonan Guo, Jian Liu, and Yingying Chen. Fitcoach: Virtual fitness coach empowered by wearable mobile devices. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2017.

[11] Men's Health. https://www.menshealth.com.

[12] A Kylasov and S Gavrov. Diversity of sport: non-destructive evaluation. *Encyclopedia of Life Support Systems*, 2:462–491, 2011.

[13] I-Min Lee, Howard D Sesso, and Ralph S Paffenbarger Jr. Physical activity and coronary heart disease risk in men: does the duration of exercise episodes predict risk? *Circulation*, 102(9):981–986, 2000.

[14] Paulo Lopez-Meyer, Stephen Tiffany, Yogendra Patil, and Edward Sazonov. Monitoring of cigarette smoking using wearable sensors and support vector machines. *IEEE Transactions on Biomedical Engineering*, 60(7):1867–1872, 2013.

[15] Mehdi. 10 bench press mistakes that kill and injure lifters. https://stronglifts.com/bench-press/mistakes/, 2017.

[16] Sean Nalewanyj. 7 side lateral raise mistakes to avoid for bigger side delts. http://seannal.com/articles/training/side-lateral-raise-mistakes.php, 2017.

[17] Daniele Riboni and Claudio Bettini. Cosar: hybrid reasoning for context-aware activity recognition. *Personal and Ubiquitous Computing*, 15(3):271–289, 2011.

[18] Chenguang Shen, Bo-Jhang Ho, and Mani Srivastava. Milift: Efficient smartwatch-based workout tracking using automatic segmentation. *IEEE Transactions on Mobile Computing*, 17(7):1609–1622, 2018.

[19] Sheng Shen, He Wang, and Romit Roy Choudhury. I am a smartwatch and i can track my user's arm. In *Proceedings of the 14th annual international conference on Mobile systems, applications, and services*, pages 85–96. ACM, 2016.

[20] Fred Stellabotte and Rachel Straub. *Weight Training Without Injury: Over 350 Step-by-Step Pictures Including What Not to Do!* Regalis Publishing, 2016.

[21] Karanbir S Toor, Ameet S Toor, Charlton M Smith, and Alexander G Orozco. Oli, your weight-training assistant. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 184–189. ACM, 2017.

[22] Nick Tumminello. Dumbbell shoulder press: Common form mistakes & corrections. http://nicktumminello.com/2015/06/dumbbell-shoulder-press-common-form-mistakes-corrections, 2015.

[23] Chuyu Wang, Jian Liu, Yingying Chen, Lei Xie, Hong Bo Liu, and Sanclu Lu. Rf-kinect: A wearable rfid-based approach towards 3d body movement tracking. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(1):41, 2018.

[24] Koji Yatani and Khai N Truong. Bodyscope: a wearable acoustic sensor for activity recognition. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 341–350. ACM, 2012.

[25] Han Zhou, Yi Gao, Xinyi Song, Wenxin Liu, and Wei Dong. Limbmotion: Decimeter-level limb tracking for wearable-based human-computer interaction. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(4):1–24, 2019.