IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 5 January 2024, accepted 3 February 2024, date of publication 6 February 2024, date of current version 14 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3363437

RESEARCH ARTICLE

uLift: Adaptive Workout Tracker Using a Single Wrist-Worn Accelerometer

JONGKUK LIM[1], **YOUNGMIN OH**[2], **(Member, IEEE), AND YOUNGGEUN CHOI**[1]
[1]Department of Computer Engineering, Dankook University, Yongin-si 16890, South Korea
[2]School of Computing, Gachon University, Seongnam-si 13120, South Korea

Corresponding author: Younggeun Choi (younggch@dankook.ac.kr)

This work was supported by the Dankook University, in 2022.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT The emergence of wearable devices has motivated people to actively log their daily exercise routines using smart apps. However, most current exercise trackers focus on aerobic exercises, and thus provide limited functionality for tracking and analyzing anaerobic workouts involving complex and repetitive movements. To fill this gap, we developed uLift, an adaptive workout tracker that uses only a single wrist-worn accelerometer and has four main functions: workout detection, repetition counting, workout classification, and quality assessment. First, uLift detects a binary workout state from continuous signals using the weighted sum of autocorrelation. Second, repetition counting is conducted by filtering out unwanted peaks. Third, the segments of a workout are used to generate a representative template for workout classification using the distances calculated from dynamic time warping. Finally, to assess workout quality, the form score is computed by evaluating the consistency across repetitions. As uLift does not require a training process, it can easily add new workouts or delete existing ones using an instant adaptation process. For the evaluation of uLift, we collected a multi-joint workout dataset comprising 15 workouts from 35 participants in a gymnasium. To allow for natural and individual variability, we provided the participants with minimum instructions. The dataset was open-sourced to facilitate future research on anaerobic workout analysis. As a result, uLift achieved 93.09% accuracy for workout detection, mean counting error of 0.61, and classification accuracy of 90.06%. The form score was significantly different among the three subgroups of participants, divided by workout experience.

INDEX TERMS Anaerobic workout, autocorrelation, classification algorithms, dynamic time warping, inertial measurement unit, quality assessment, repetition counting, wearable, workout tracker.

I. INTRODUCTION

Regular exercise is essential for promoting health and improving the quality of life [1], [2], [3], [4], [5], [6], [7]. Additionally, regularly logging one's exercise can encourage the continuation of workout activities. In line with these trends, smart wearables have started adding tracking functions that motivate users to exercise regularly [8], [9], [10], [11], and many people have begun to adopt this new technology to keep track of their workouts [12], [13]. However, smartwatches or smartbands with motion tracking and logging functions are usually designed for aerobic exercises, such as walking, running, and cycling. Although anaerobic exercises or weight training, which help strengthen body muscles, are critical for maintaining health in conjunction with aerobic exercise [14], existing solutions only provide limited functionality for automatic logging and systematic analysis of anaerobic exercises.

Therefore, we propose uLift, an adaptive anaerobic workout tracker that automatically logs and analyzes anaerobic exercises using a single wrist-worn accelerometer. There are four main functions that uLift provides: (1) detecting a

The associate editor coordinating the review of this manuscript and approving it for publication was Lorenzo Mucchi.

user's workout state, (2) counting repetitions, (3) classifying workout types, and (4) assessing the quality of workouts, all with minimal user intervention. Figure 1 presents a flow diagram of uLift. To enable anaerobic workout tracking without user intervention, the system must be able to detect whether the user is in a workout state. To detect the workout state, we first defined anaerobic workouts as actions involving repetitive movements to exhaust target muscles. Therefore, the workout state can be detected by first recognizing consistent repetitions from sensor values. When uLift determines that the user is in a workout state, it counts the number of repetitions, classifies the workout type, and finally assesses the workout quality. Two challenges must first be overcome before the workout tracker can be practically applicable to real-world scenarios. First, users must be able to add custom workout types or delete the existing ones, based on their needs and progress. Second, the system must deal with the intraclass variability caused by variations from the same user performing the same exercise in different manners, which can also drift with time. Therefore, the system must adapt to these changes in the workout types and motions. uLift can accomplish such tasks via straightforward signal processing techniques, such as autocorrelation and dynamic time warping, and does not rely on data-driven parameter updates as machine learning algorithms do. The absence of a learning process enables the seamless addition or deletion of custom work types. The adaptability of uLift, along with its simple unisensory configuration on the wrist, makes it a practical solution for anaerobic workout tracking and logging.

The main contributions of the current study are summarized as follows:

- **Dataset of real-world multi-joint workouts**: To evaluate uLift, we collected and open-sourced a dataset[1] featuring 15 types of multi-joint workouts from 35 participants in a real gym environment. This dataset is intended to facilitate future research and development in the field of anaerobic workout analysis while offering an extended diversity compared with existing datasets [15], [16], [17], [18].
- **Automatic workout analysis without the need for training**: We present a novel method for fully automatic workout analysis. This comprehensive approach encompasses the detection of workout states through repetitive motion recognition, repetition counting, adaptable workout type classification, and workout form quality assessment. Notably, uLift does not require a traditional training process, thereby enhancing its ease of deployment, and offering an advantage over the predominantly data-driven approaches in the existing literature [15], [19], [20], [21], [22], [23], [24], [25], [26]. uLift also yields consistent and robust performance against the change of parameters and thresholds.

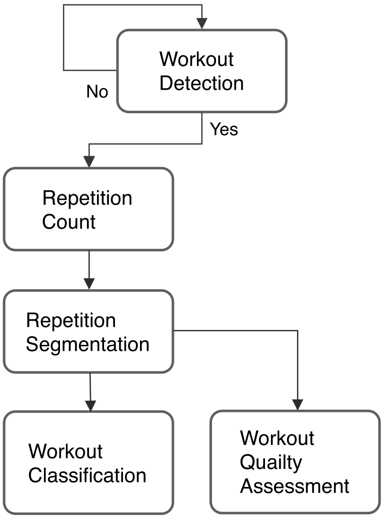


FIGURE 1. **Flow diagram of uLift workout analysis.**

- **Algorithm for personalized workout classification**: Our method employs a template-based system that requires only a single action instance from the user for personalized workout classification. Designed for flexibility, it allows for the easy addition or removal of workout types to accommodate the individual's motion variations and reflect changes in condition and skill level. uLift surpasses the previous automatic workout analysis algorithms [15], [19] as well as the deep learning models [20], [21] particularly when the number of available templates is small, which is crucial for a personalized, adaptive workout tracker. Unlike these methods that typically rely on a training process and may take time to adapt to new data, uLift can immediately incorporate new workouts, offering a significant advantage in dynamic and personalized workout scenarios.
- **Assessment of workout quality without human annotation**: Diverging from conventional methods that often rely on subjective human annotations [27], [28], [29], [30], [31], our method assesses workout form quality by examining the stability of motion. Based on the premise that skilled individuals perform their workouts with consistent movements, this approach offers a practical way to evaluate workout consistency and form, providing insights into the overall quality of the exercise.

The remainder of this paper is organized as follows. Section II reviews the previous studies on automatic workout analyses. Section III describes the experimental process and collected dataset as well as the methodologies to implement- automatic workout motion analysis. Section IV presents the results of applying the four methodologies to the collected dataset described in Section III-A. Finally, Section V concludes the paper with a discussion of the results.

[1] https://github.com/JeiKeiLim/uLift-dataset

II. RELATED WORK

A. SEMI-AUTOMATIC WORKOUT ANALYSIS

Unlike a fully automatic workout analysis, which is discussed in Section II-B, a semi-automatic workout analysis requires manual user input. A primary example is the requirement for users to manually indicate the beginning and end of their workout sessions. Nonetheless, many studies in this category provide workout classification, repetition counting, and optional workout quality assessments using wearable inertial sensors. However, the requirement for user intervention during workouts inevitably lowers the usability of practical workout-tracking systems. Hosein et al. [22] performed workout classification using feature extraction, and compared the results from multiple classifiers. Ebert et al. [23] utilized multiple inertial measurement units (IMUs) to classify workout types and compared the results from different combinations of attached sensor locations. Their results indicated that using a single sensor attached to the wrist resulted in a classification accuracy comparable to that of other combinations. Margarito et al. [32] applied dynamic time warping (DTW) to classify workout types. However, they focused on aerobic workouts such as walking, yoga, and cycling. Zhou et al. [33] adopted a pressure array sensor to classify the movement type, count the number of repetitions, and estimate quality. Hussain et al. [24] applied a long short-term memory (LSTM) neural network to classify 42 exercises targeting six muscle groups. Because the entire session lasted six weeks, only four subjects were able to complete it. Coates and Wahlström [28] introduced LEAN, which simultaneously classifies and estimates the quality of three free-weight (dumbbell) workouts in addition to measuring the range of motion (ROM). However, they tested their algorithms on only three isolated muscle movement sets.

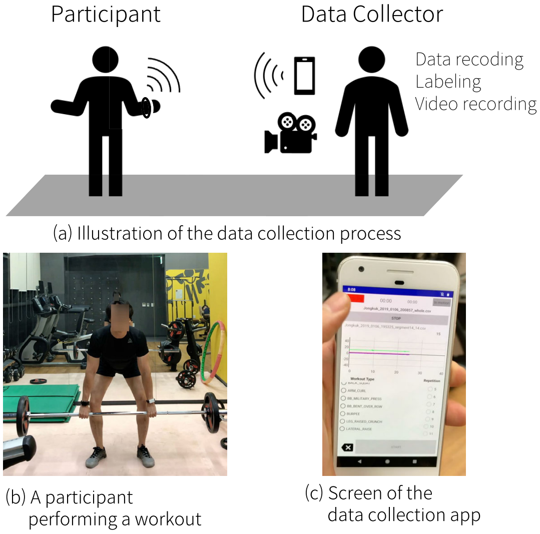
B. FULLY AUTOMATIC WORKOUT ANALYSIS

To achieve a fully automatic workout analysis without user intervention, the system must be able to detect the beginning and end of a workout, classify the workout type, and count repetitions. One of the pioneers of automatic workout analysis was RecoFit [15], which uses a support vector machine (SVM) to recognize the state and type of workout. Repetition counting was performed by extracting the peak candidates and filtering out false positives based on the known minimum and maximum time parameters for a single repetition of the recognized workout. This approach requires prior knowledge of the individual repetition time range of each workout, which complicates the addition of custom workout types. Following the successful study by RecoFit [15], MiLift [19] proposed the use of autocorrelation and revisited algorithms for workout state recognition. The repetition counts were reliably tracked by detecting the appropriate peaks and valleys. However, this method has limitations when applied to complex multi-joint workout motions where two or more peaks may appear in a single motion. In addition, most of the movements were conducted as isolation workouts, which might result in lower accuracy of the multi-joint workout motion. FitCoach [25] attempted to identify the similarities between the movements of regular users and trainers. Additionally, the workout state was detected using the peak number of autocorrelations. However, an autocorrelation peak may not occur when one repetition of the workout is prolonged. For workout-type classification, both MiLift [19] and FitCoach [25] utilize RecoFit's [15] approach with customized features. Kowsar et al. [26] applied an unsupervised algorithm to detect and count the repetitions of new workout routines. However, they tested the algorithm for a relatively small real gymnasium setup with two subjects performing 11 workouts. Ishii et al. [34] presented ExerSense which segments, classifies, and counts exercises using a correlation-based method and DTW. The authors also tested different device locations to achieve the best performance. However, their study was limited to testing five exercises, including both aerobic and anaerobic workouts (running, walking, jumping, push-ups, and sit-ups). Džaja et al. [35] utilized the frequency spectrum of acceleration magnitude for workout segmentation and DTW for classification, using data from three body-worn IMU sensors. They applied this algorithm to nine anaerobic workouts.

These studies showed that combining workout state detection, repetition counting, and workout-type classification can successfully track a user's anaerobic workouts. However, tracking workouts using machine learning has limitations in that only pre-trained workout types can be tracked. Adding and deleting workout types is an essential requirement for practical use because new workout types can be defined by users. However, studies employing non-machine learning algorithms often focus on a limited range of workouts and subjects, which may leave their generalizability more open to exploration and confirmation.

C. WORKOUT QUALITY ASSESSMENT

The effect of anaerobic workouts depends on how well the posture is maintained. Therefore, it is essential to assess the correctness of a user's workout format. Velloso et al. [27] measured the workout form quality of a user through four sensors. In addition, they identified the angle of the wrong posture using a Kinect. The joint angle is a useful indicator for measuring the workout form. However, obtaining a joint angle with an inertial sensor requires multiple sensors to extract the angle, complicating its implementation in practical applications. Pernek et al. [30] proposed a method for measuring the exercise intensity using sensor data. However, measuring workout intensity with inertial sensors is challenging due to the highly subjective nature and individual variability of proper workout intensity. Kowsar et al. [31] exploited anomaly detection to identify incorrect workout motions using only correct posture data. Coates and Wahlström [28] estimated range of motion (ROM) in three free-weight exercises. Spliz and Munz [29] trained a neural network model to assign Functional Movement Screening



(a) Illustration of the data collection process

(b) A participant performing a workout

(c) Screen of the data collection app

FIGURE 2. Experiment setup.

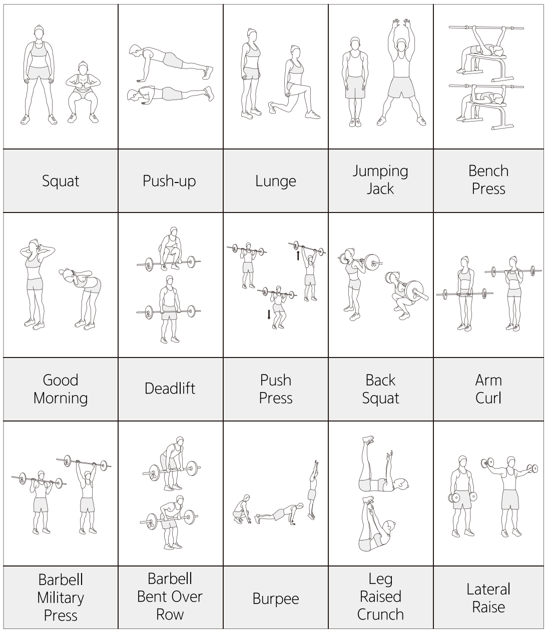


FIGURE 3. Workout types.

scores to four exercises with 17 inertial measurement units.

Assessing workout quality is a challenging problem because the definition of 'correct form' is inherently ambiguous, varying with the workout's context and individual differences. Furthermore, defining the correct form necessitates manual annotation by human experts, a process that is both costly and challenging to scale for diverse workouts. Additionally, estimating human poses with inertial sensors is often noisy and unreliable in many scenarios. Therefore, we designed uLift to indirectly assess workout quality by measuring the consistency of each repetition during a workout.

III. METHOD

A. DATASET

1) HARDWARE

To collect workout data, we developed a custom-designed wristband and an Android data collection application (Figure 2). We used the IMU unit of BNO055 [36] and the Bluetooth module of nRF51822 [37] for Bluetooth low energy (BLE) communication. Given that BLE, a wireless communication method, was used for interaction with the data collection application, the sampling rate varied and averaged around 60 Hz. We utilized a single accelerometer sensor for data collection because multiple studies [18], [22], [23], [30], [38] have shown desirable results using a single accelerometer sensor on the wrist.

2) WORKOUT ENVIRONMENT

We collected data from 15 distinct multi-joint workout types, chosen over isolated workouts to capture complex and more realistic data. Figure 3 shows the 15 selected workouts: Squat, Push-up, Lunge, Jumping Jack, Bench Press, Good Morning, Deadlift, Push Press, Back Squat, Arm Curl, Barbell Military Press, Barbell Bent-over Row, Burpee, Leg-raised Crunch, and Lateral Raise. To collect workout data as close to an unconstrained environment as possible, we asked the participants to exercise as usual. They were informed of the standard workout motion only when they had no previous knowledge of the exercise. Additionally, for workouts in which hand movements could be performed differently, such as squats, lunges, leg-raised crunches, and burpees, the participants were requested to perform in a manner that was comfortable for them. Consequently, not only did we observe variations in hand movements among different participants performing the same workout, but we also noted inconsistencies within the same individual's execution of the exercise across attempts.

To clarify the analysis framework, we define a 'session' as the period from the start to the end of data collection. Each session comprises 'workout segments' and 'rest segments'. During each session, participants were asked to complete 15 different types of workouts and perform repetitions according to their condition in each 'workout segment.' Consequently, while the target was 10 repetitions per workout, we observed variations ranging from fewer than eight to more than 12 repetitions, reflecting the diverse conditions of participants.

3) PARTICIPANTS

35 volunteers participated in this study and all of them signed the informed consent prior to participation. Their ages ranged from 20 to 35 years, with male and female percentages of 72% and 28%, respectively. To ensure diversity in workout experience, we recruited people with different workout experiences, from those without any experience to those

TABLE 1. Data collection Table.

# of workout types	15
# of participants	35
# of sessions	158
# of workout segments	2,355
# of rest segments	2,552
# of repetitions	23,738
Total time of data collection	40 Hours
Total time of workout segments	14 Hours
Total time of rest segments	26 Hours

with up to five years of workout experience. The maximum number of sessions allowed per day for each participant was two, with up to six sessions per person. The total number of collected sessions was 158, and the total number of exercise repetitions was 23,738. Segment annotation for the beginning and end of the workout was performed directly by the experimenters through a data collection application, and video recording was conducted to adjust the labeling details. The total data collection duration was approximately 40 h. After each workout segment, participants were allowed as much rest time as required through the resting segments. Furthermore, we did not provide any instructions on what to do during the rest segment, which led the participants to perform natural actions such as walking, drinking water, intermittent running, and stretching. Table1 summarizes the collected data.

B. WORKOUT DETECTION

To recognize user engagement in a workout, it is essential to establish a criterion for the workout motion. Anaerobic workout motion is defined as the act of exhausting the target muscle by intentionally repeating the same movements. Detecting the workout state thus requires determining whether the same movement is performed repeatedly. From the perspective of an inertial sensor, this translates to identifying the periods during which a consistent signal is observed repeatedly.

1) COMPUTING SIGNAL PERIOD

Assuming repetitive movements, autocorrelation can be utilized to determine the periodic rate. While typically one axis is chosen for autocorrelation analysis in a three-axis accelerometer sensor [32], this study employed a weighted sum across all three axes after the autocorrelation analysis. Furthermore, to ensure consistent data size for autocorrelation, a sliding window method was adopted.

$$S(\boldsymbol{X}) = \sum_{i}^{i=x,y,z} \text{ACF}(\boldsymbol{X}_i) \cdot \frac{exp^{||\text{ACF}(\boldsymbol{X}_i)||}}{\sum_{j}^{j=x,y,z} exp^{||\text{ACF}(\boldsymbol{X}_j)||}} \quad (1)$$

Equation (1) defines $S(\boldsymbol{X})$, a function that calculates the weighted sum of autocorrelations (ACFs) along the three axes, x, y, and z. Here, $\text{ACF}(\boldsymbol{X}_i)$ signifies the autocorrelation analysis performed independently on each axis i. By utilizing a softmax function, each axis is assigned a weight derived from its respective ACF. This methodology facilitates the extraction of stable autocorrelation values amidst complex multi-joint movements, a phenomenon depicted in Figure 4. Subsequently, the determined weighted autocorrelation value becomes a crucial parameter in identifying the periodicity of the repetitive signal:

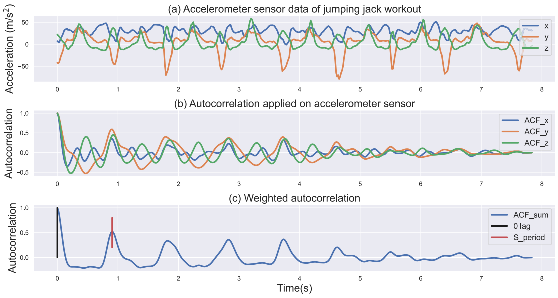


FIGURE 4. Weighted autocorrelation of jumping jack workout (a) Raw accelerometer data from workout segment, (b) Autocorrelation applied on the raw accelerometer sensor data, (c) Weighted autocorrelation using 3-axis of autocorrelation from (b).

$$\tau(W) = \frac{P(S(\boldsymbol{X}[k - W + 1 : k]))}{H} \quad (2)$$

where $\tau(W)$ represents the computed signal period derived from a window W applied to the signal $\boldsymbol{X}$. The function P denotes the extraction of the period, or peak index, from the weighted autocorrelation function $S(\boldsymbol{X})$. It is important to note that H represents the sampling rate (in Hz) of the sensor, providing a scalar divisor to normalize the period. Equation (2) forms a foundation for establishing the periodic characteristics of the signal $\boldsymbol{X}$ in the ensuing signal processing.

2) WORKOUT STATE RECOGNITION

Using the signal period of the repetitive motions as determined by the method described in the previous section, the binary workout state can be detected. During repetitive motion segments, the signal period remains relatively stable. In contrast, during segments with irregular movements, the signal period experiences rapid fluctuations. Therefore, consistent signal periods are indicative of anaerobic workouts, whereas segments with rapidly fluctuating signal periods typically correspond to non-exercising intervals, such as rest segments.

To effectively analyze workout states based on these derived characteristics, the computed signal periods τ are aggregated into a vector, denoted as $\mathbf{SP}(W)$:

$$\mathbf{SP}(W) = [\tau_n, \tau_{n-1}, \cdots, \tau_{n-W+1}] \quad (3)$$

where each τ_n denotes a signal period, calculated with a window W of the signal $\boldsymbol{X}$, culminating at the n^{th} sample. In a more explicit formulation, each τ_n reflects a localized measure of the signal period, based on the W samples of $\boldsymbol{X}$ terminating at the n^{th} sample. This stacking of periods

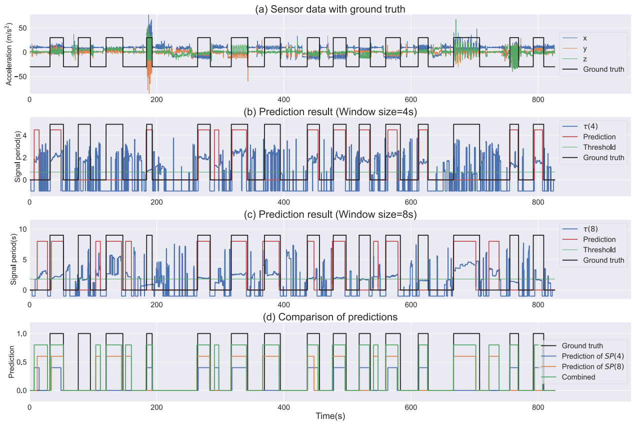


FIGURE 5. Predictions by $\mathrm{SP}(W_4)$ and $\mathrm{SP}(W_8)$: (a) Raw sensor data and ground truth of workout state, (b) Prediction result by $\mathrm{SP}(W_4)$, (d) Prediction result by $\mathrm{SP}(W_8)$, (e) Comparison of each prediction and combined prediction.

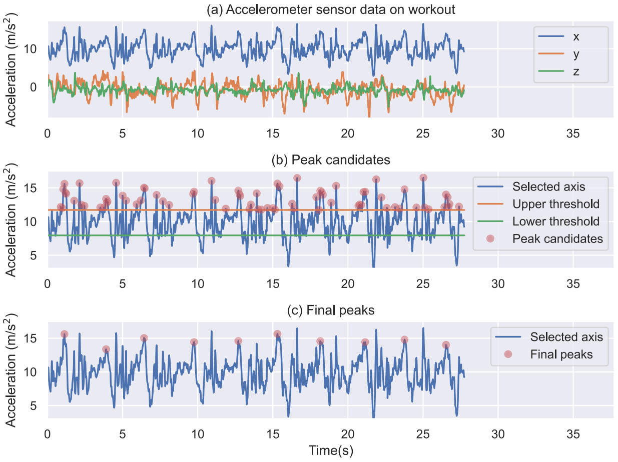


FIGURE 6. Finding final peaks from peak candidates: (a) Raw sensor data, (b) Peak candidates and threshold, (c) Final peaks.

in $\mathbf{SP}(W)$ is integral to developing an understanding of the temporal consistency or variations within the repetitive signal being analyzed.

To identify the segments where the previously defined anaerobic exercise movements occurred, we examined whether there were intervals where the value of $\mathbf{SP}(W)$ was above a certain threshold for a specific duration. In this study, we detected both quick and slow repeated exercise motions by calculating two $\mathbf{SP}(W)$ values with different sliding window lengths. The lengths of the sliding windows were set to 4 and 8 s for the fast and slow motions, respectively, and the $\mathbf{SP}$ values using these lengths were denoted as $\mathbf{SP}(W_4)$ and $\mathbf{SP}(W_8)$, respectively.

These lengths were selected based on observations of the exercise data, which indicated that the majority of single repetitions were completed within 4 s. However, we also noted occurrences of movements extending beyond 4 s, such as burpees. Consequently, a 4-s window was employed to detect rapid workouts, while an 8-s window was utilized for exercises with prolonged duration.

After computing $\mathbf{SP}(W_4)$ and $\mathbf{SP}(W_8)$ with window sizes of 4 and 8 s, respectively, any $\tau(W)$ values exceeding 0.7 in $\mathbf{SP}(W_4)$ and 1.8 in $\mathbf{SP}(W_8)$ were considered to be in the workout state (Figure 5). Our rationale for using the threshold of 0.7 in $\mathbf{SP}(W_4)$ is that one repetition cannot be completed in less than 0.7 s. A threshold of 1.8 in $\mathbf{SP}(W_8)$ was established to address instances where $\tau(W)$ might not be accurately captured for movements extending beyond 2 s with the sliding window size used in $\mathbf{SP}(W_4)$. In scenarios where repeated movements are consistent and stable, a 4-s window may suffice to correctly extract $\tau(W)$ for actions exceeding 2 s. Conversely, for exercises performed with irregular and unstable movements, a longer window, specifically the 8-s window of $\mathbf{SP}(W_8)$, is required to ensure accurate detection of prolonged repetitions.

Our thresholds for $\mathbf{SP}(W_4)$ and $\mathbf{SP}(W_8)$ were designed to optimize the detection accuracy for both fast and slow movements, and refining the interpretation of the recognized states could enhance the overall robustness. Consequently, intervals that are recognized as the workout state and are shorter than a predetermined duration are dismissed. Furthermore, if the workout state was identified again within a specific duration, it was interpreted as a continuation of the previous state.

C. REPETITION COUNTING AND WORKOUT SEGMENTATION

In principle, the number of repetitions within a workout segment can be determined by counting the signal peaks during the active workout state. However, this approach encounters challenges due to the variability in real-world data. For instance, complex workout motions might result in more than two peaks within a single repetition, as illustrated in Figure 6. To address this, our methodology filters out extraneous peaks by utilizing what we refer to as the 'workout rate', the representative signal period within the workout segment.

It is crucial to differentiate between 'workout segments' as defined in our dataset, specific periods of exercise activity, and 'workout segmentation,' a distinct methodology we introduce here. 'workout segmentation' is the analytical process of dissecting sensor data into individual repetitions within each workout segment.

1) COMPUTING WORKOUT RATE

The workout rate in the workout segment is determined using the signal period $\mathbf{SP}$ as the following equation.

$$\mathrm{WR} = \frac{\eta_{45}(\mathbf{SP}(W)) + \eta_{95}(\mathbf{SP}(W))}{2} \tag{4}$$

where WR denotes the workout rate within the segment; the terms η_{45} and η_{95} represent the 45th and 95th percentiles of $\mathbf{SP}$, respectively. During our observations of the workout movements, we noted slight variations in τ during the

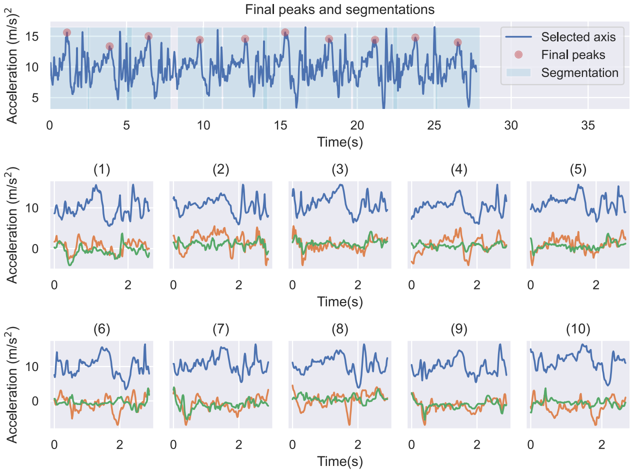


FIGURE 7. **Final peaks and repetition segmentation.**

initial phases. As the workout progresses to its middle and later phases, a more consistent τ value emerges, possibly because individuals find a rhythm or pace that suits them. To best represent the typical movement pattern throughout the workout and minimize the influence of minor variations, particularly during the initial phases, we employed the 45th and 95th percentiles of **SP**. These percentiles encompass a broad range of central τ values, filtering out the extreme values that may not indicate the overall workout pattern.

2) FINDING FINAL PEAKS USING PEAK CANDIDATES

Once the workout rate WR was obtained, the axis with the largest amplitude was selected from the workout segment. The reason for choosing the axis with the largest amplitude, and not the weighted sum of autocorrelation, is to utilize the unaffected sensor data to extract the most appropriate peak candidates. Afterward, among the detected peaks, those below 80% of the percentile in amplitude were filtered out. The remaining peaks were regarded as peak candidates, and the final peaks were determined after the peaks within half the time of the workout rate $\frac{\mathrm{WR}}{2}$ were eliminated, as illustrated in Figure 6(b) and Figure 6(c).

3) REPETITION SEGMENTATION

In general, to conduct segmentation, it is necessary to first recognize the start and the end point of each workout repetition. However, it is usually difficult to clearly distinguish them from noisy and complex sensor data. Therefore, in this study, one segment was defined as from and to the time corresponding to half the workout rate around the final peak points obtained in the previous section.

$$s_i = [x_{p_i - H\frac{\mathrm{WR}}{2}}, x_{p_i + H\frac{\mathrm{WR}}{2}}] \tag{5}$$

Figure 7 shows the segmentation using $\frac{\mathrm{WR}}{2}$ time to the final peak center.

D. WORKOUT CLASSIFICATION

1) ADDING CUSTOM WORKOUT TYPE

Numerous studies [15], [19], [23], [25] have employed machine-learning models to classify workout types and have achieved reliable results. However, there are cases where equivalent exercises are conducted differently over time by the same user. Furthermore, different users could perform the same workout differently. Therefore, a practical solution to real-world workout classification must address the motion variability. In addition, the classifier must allow the addition of new workout types when necessary. In machine-learning methods, to add a new workout type or embrace a variation of a workout, the model must be retrained before being fetched, which is expensive and impractical. To solve the limitations of machine learning methods, this study proposes a method that does not involve learning, so that our method can add or delete workout types and adapt to workout variability.

2) GENERATING TEMPLATE DATA

As multiple segments were generated from a single workout through the process of workout segmentation, the remaining issue was the selection of the segment for classification, as this choice could significantly influence accuracy. One straightforward approach might be to randomly choose a segment or select the first one. However, this method does not guarantee the selection of the segment most representative of the entire set. Moreover, averaging all segments could lead to inaccuracies, as certain segments might contain sharp peaks or deformations that skew the overall representation. Therefore, as shown on the left side of Figure 8, dynamic time warping (DTW) [39] was conducted for the number of combinations ${}_nC_2$ in all segment pairs.

DTW in (6) computes the similarity between two time-series sequences and the optimal alignment path specifying the correspondence of points between the sequences.

$$\begin{aligned}(\mathcal{D}(n,m), \mathcal{P}^{(n,m)}) &= \mathrm{DTW}(s_n, s_m)\\ \mathcal{P}^{(n,m)} &= (\mathcal{P}_n^{(n,m)}, \mathcal{P}_m^{(n,m)})\end{aligned} \tag{6}$$

where $\mathcal{D}(n,m)$ and $\mathcal{P}^{(n,m)}$ represent the distance and the optimal alignment path between segments s_n and s_m computed by DTW. The optimal alignment path has two components: $\mathcal{P}_n^{(n,m)}$ is the optimal alignment path of s_n with respect to s_m and $\mathcal{P}_m^{(n,m)}$ is the optimal alignment path of s_m with respect to s_n. After performing DTW for each pair, the pair with the lowest Euclidean distance was chosen. The equivalent process was then conducted on the pair with the second-lowest distance to generate an average segment by (7) and (8).

$$\begin{aligned}(i,j) &= arg \min_{(n,m)\in\mathcal{C}}(\mathcal{D}(n,m))\\ \mathcal{C}' &= \mathcal{C} \setminus (i,j)\\ (p,q) &= arg \min_{(n,m)\in\mathcal{C}'}(\mathcal{D}(n,m))\end{aligned} \tag{7}$$

where (i,j) and (p,q) are the pairs with the minimum distances and the second-minimum distances, respectively.

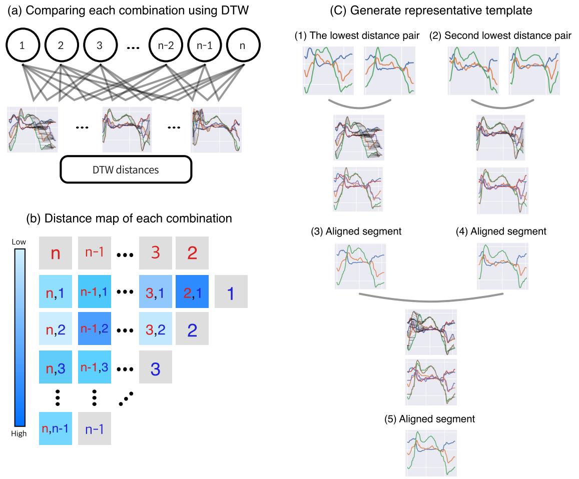

(a) Comparing each combination using DTW
1
2
3
...
n-2
n-1
n
...
...
DTW distances
(b) Distance map of each combination
Low
High
n
n-1
...
3
2
n,1
n-1,1
...
3,1
2,1
1
n,2
n-1,2
...
3,2
2
n,3
n-1,3
...
3
n,n-1
n-1
(C) Generate representative template
(1) The lowest distance pair
(2) Second lowest distance pair
(3) Aligned segment
(4) Aligned segment
(5) Aligned segment

FIGURE 8. **Generating template data (a) Computing DTW distance with each combination of the segments, (b) Distance map to extract the lowest distance pair, (c) Generating representative template using the lowest pairs.**

The functions identify these pairs by finding the minimum distances within the sets of segment combinations $\mathcal{C}$ and $\mathcal{C}'$. Then, the transform function T in (8) warps signal s into the optimal alignment path $\mathcal{P}$.

$$s_{r1} = \frac{1}{2}\left(T(s_i, \mathcal{P}_i^{(i,j)}) + T(s_j, \mathcal{P}_j^{(i,j)})\right)$$
$$s_{r2} = \frac{1}{2}\left(T(s_p, \mathcal{P}_p^{(p,q)}) + T(s_q, \mathcal{P}_q^{(p,q)})\right) \quad (8)$$

Finally, as shown on the right side of Figure 8, the two averaged segments are used to generate the representative template data:

$$R = \frac{1}{2}\left(T(s_{r1}, \mathcal{P}_{r1}^{(r1,r2)}) + T(s_{r2}, \mathcal{P}_{r2}^{(r1,r2)})\right) \quad (9)$$

3) CLASSIFYING WORKOUT TYPES WITH DTW

The representative template data produced through the earlier process were compared with pre-registered template data using DTW to calculate the distance between them. The workout type corresponding to the template with the minimum distance was then identified as the recognized workout. Nevertheless, the same individual may perform the same workout differently based on their condition on any given day. Therefore, to accommodate this intraclass variability, multiple templates for a single type of workout were adopted, as illustrated in Figure 9. Finally, the workout type (class) of a test template x is determined by:

$$\text{class}(x) = \arg\max_{k}\left(1 - \frac{e^{\frac{1}{m_k}\sum_j^{m_k}\mathcal{D}(x,y_{k,j})}}{\sum_i^n e^{\frac{1}{m_i}\sum_j^{m_i}\mathcal{D}(x,y_{i,j})}}\right) \quad (10)$$

where n is the number of currently stored workout types, m_k is the number of templates corresponding to the k^{th} workout type, and $y_{i,j}$ is the j^{th} template stored in the i^{th} workout type. Therefore, k with the highest softmax value is classified as the k^{th} workout type. In this study, the number of templates, m_k, was set to a maximum of three, as in our findings in Section IV, and every time the same workout type was performed, the stored templates were replaced with the latest templates.

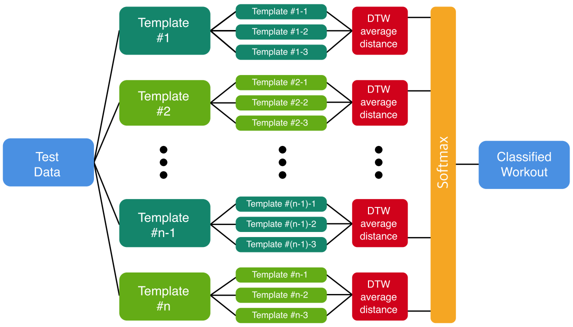


FIGURE 9. **Classifying workout type with DTW.**

E. WORKOUT QUALITY ASSESSMENT

Measuring the joint angles from inertial sensors requires attaching multiple sensors with delicate calibrations, which adds a burden to users. Furthermore, a workout form that appears incorrect may be the correct form, depending on the physical condition of the user. Therefore, in this study, the factor that measures workout quality was defined as how consistently the motion was performed for each repetition of the workout. This factor was determined based on the observation that a person with experience in a certain workout is more likely to perform it more consistently than those with less experience. Therefore, to measure the form score, the method depicted on the left side of Figure 8 was reused.

When calculating distances using DTW across paired repetition segments, we only considered the top third of the total distance when computing the form score. This approach was adopted to eliminate potential outliers within the workout movements and measure motion consistency more reliably. Although physical workouts often exhibit considerable variability between individual actions due to factors such as fatigue, technique, and individual biomechanics, the focus of this study was to gauge the consistency of the most similar workout motions. The form score $\mathcal{F}$ is given by:

$$\mathcal{D}_{\text{all}} = \{\mathcal{D}(n_1, m_1), \mathcal{D}(n_2, m_2), \cdots, \mathcal{D}(n_{|\mathcal{C}|}, m_{|\mathcal{C}|})\}$$
$$\mathcal{F} = \frac{1}{\lfloor\frac{|\mathcal{C}|}{3}\rfloor}\sum_{i=1}^{\lfloor\frac{|\mathcal{C}|}{3}\rfloor}\text{sort}(\mathcal{D}_{\text{all}})[i] \quad (11)$$

where $|\mathcal{C}|$ and $\mathcal{D}_{\text{all}}$ represents the total length and the collection of all DTW distances for the pairs (n, m) in the segment combination $\mathcal{C}$, respectively. The notations $n_1, m_1, n_2, m_2, \cdots, n_{|\mathcal{C}|}, m_{|\mathcal{C}|}$ specify individual pairs within the set. Each pair (n_i, m_i) indicates a unique comparison between specific repetitions. Consequently, a lower value of $\mathcal{F}$ indicates consistent motion quality in the workout segment, while a higher value suggests inconsistency.

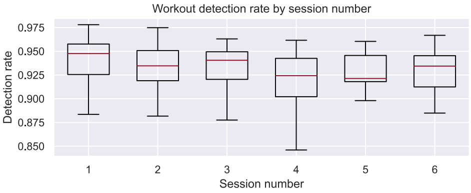


FIGURE 10. **Workout detection rate by session numbers.**

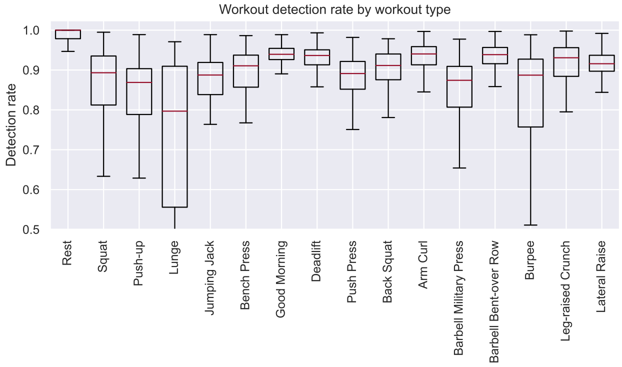


FIGURE 11. **Workout detection rate by workout types.**

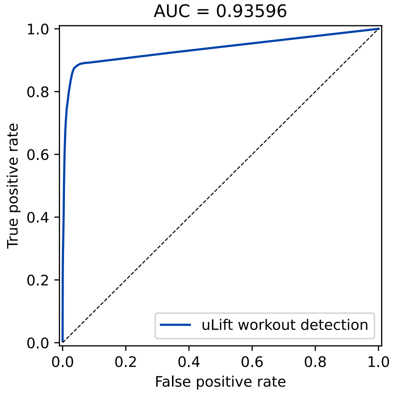


FIGURE 12. **Receiver operating characteristic (ROC) curve.**

TABLE 2. **Error rate of repetition.**

Mean accuracy	Mean error repetition
93.87%	0.61

IV. EVALUATION

This section evaluates the accuracy and performance of the methods described in Section III by applying them to the collected dataset introduced in Section III-A. Evaluations included workout detection, repetition counting, classification, and quality assessment.

A. WORKOUT DETECTION

Ensuring the ability of a workout detection system to detect a broad spectrum of workout types reliably is fundamental for its utility and efficacy in real-world applications. As we aim to ascertain the robustness of the uLift system in this context, the subsequent analysis provides a detailed exploration of its performance metrics across varied session numbers, as well as its proficiency in identifying different types of workouts.

We defined the detection rate as the proportion of correctly detected time points out of those in the entire session and used it as a metric for workout detection. We evaluated the detection rates separately for individual sessions in which the subjects participated (Figure 10). The global mean detection rate was 93.09% and exceeded 92% in all individual sessions, showing robustness to intra- and inter-class variability among subjects.

In addition, workout detection rates were calculated for each workout type (Figure 11). The results demonstrated stable detection rates for the majority of workout types. A relatively low detection rate with higher variability was observed for lunge motions, possibly because of the inherent asymmetry in lunge movements. Furthermore, because only minimal instructions were given to the participants regarding the execution of the workout motions, some subjects changed their arms mid-motion during the lunge exercises. It is anticipated that instructing users to maintain consistent arm movements will lead to improved results.

Detection of a workout state was determined from the two thresholds, 0.7 for $\mathbf{SP}(W_4)$ and 1.8 in $\mathbf{SP}(W_8)$, as described in section III-B. To test the sensitivity of the detection rate on different thresholds, a constant ranging from -1 to 3 was multiplied to both thresholds, and the resulting false positive rates and the true positive rates were calculated (Figure 12). The estimated area under the curve was 0.936, indicating the robust performance of uLift on workout detection.

The proposed method is characterized by its ability to detect repetitive motions inherent in activities such as anaerobic workouts and anticipate stable performance across different types of workouts. Notably, the design of the method, which does not rely on a training process, suggests its potential to robustly detect workout states even when presented with previously unseen workout motions.

B. REPETITION COUNTING

The number of repetitions of the entire workout dataset was measured using the proposed method and the estimated average error was 0.61 per single workout session, which consisted of approximately 10 repetitions.

Figure 13(a) shows the average number of errors for each workout. Relatively high error rates were observed for Jumping Jack, Burpee, and Lunge. For the Jumping Jack, the average movement speed was faster than that in other workouts, and errors could have occurred while the user was jumping up and down, leading to a disturbance in the balance of the user. For Burpee and Lunge, errors could occur because the participants often became distracted by the difficulty in maintaining the body's center of gravity. In particular, the lunge is more likely to cause errors because of its asymmetrical movements. Figure 13(b) presents the

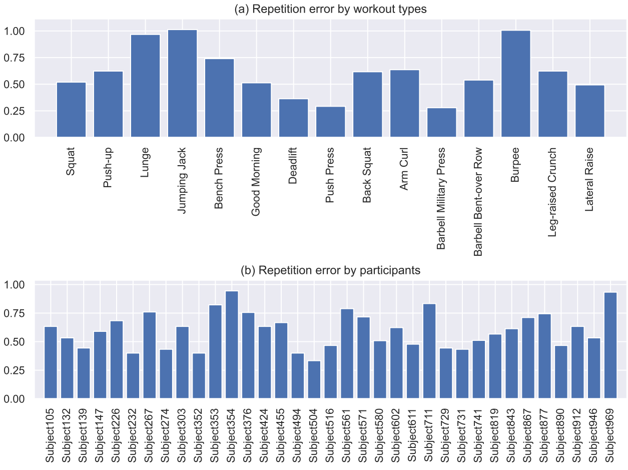


FIGURE 13. **Repetition error by workout type and participants (a) Repetition error by workout types (b) Repetition error by participants.**

TABLE 3. **Workout classification accuracy.**

# of templates	uLift	RecoFit	Milift	CNN	LSTM
1	**87.9%**	79.5%	81.2%	69.2%	71.6%
2	**90.0%**	80.2%	81.0%	74.2%	80.7%
3	**90.0%**	80.1%	85.7%	87.1%	89.3%
4	88.5%	83.3%	85.3%	85.7%	**91.6%**
5	89.8%	84.4%	88.0%	**94.2%**	92.0%
Average	**89.2%**	81.7%	84.2%	82.1%	85.0%

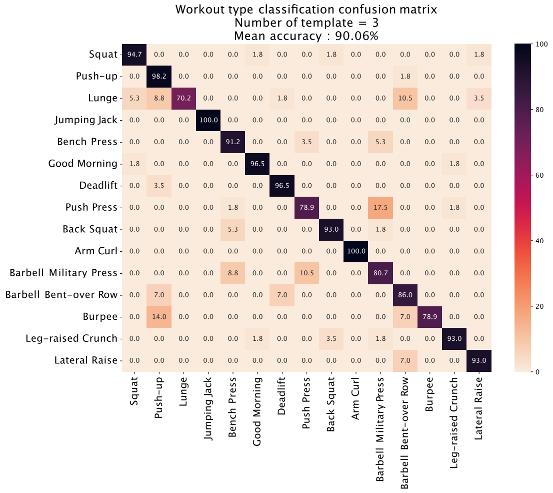
TABLE 4. **Workout classification F1-score.**

# of templates	uLift	RecoFit	Milift	CNN	LSTM
1	**0.880**	0.768	0.786	0.635	0.680
2	**0.900**	0.772	0.785	0.692	0.780
3	**0.901**	0.770	0.831	0.847	0.877
4	0.887	0.793	0.825	0.832	**0.905**
5	0.900	0.802	0.847	**0.923**	0.897
Average	**0.894**	0.781	0.815	0.786	0.828

average number of errors for each participant. The maximum error was less than one for all participants despite their diversity in workout experiences (Section IV-D). According to a user survey conducted by MiLift [19], the number of errors in our study is in the range of the maximum number of errors tolerable by users.

C. WORKOUT CLASSIFICATION

The same workout performed by the same person can vary depending on the body condition on the day, muscle fatigue accumulated with repetition, or simple progress with practice. Therefore, it is essential for a workout classifier to adapt to the variability in workout patterns. Accordingly, uLift uses multiple templates from the same workout. To verify the optimal number of templates, classification experiments were conducted by changing the number of templates from one to five.



	Squat	Push–up	Lunge	Jumping Jack	Bench Press	Good Morning	Deadlift	Push Press	Back Squat	Arm Curl	Barbell Military Press	Barbell Bent–over Row	Burpee	Leg–raised Crunch	Lateral Raise
Squat	94.7	0.0	0.0	0.0	0.0	1.8	0.0	0.0	1.8	0.0	0.0	0.0	0.0	0.0	1.8
Push–up	0.0	98.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.8	0.0	0.0	0.0
Lunge	5.3	8.8	70.2	0.0	0.0	0.0	1.8	0.0	0.0	0.0	0.0	10.5	0.0	0.0	3.5
Jumping Jack	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Bench Press	0.0	0.0	0.0	0.0	91.2	0.0	0.0	3.5	0.0	0.0	5.3	0.0	0.0	0.0	0.0
Good Morning	1.8	0.0	0.0	0.0	0.0	96.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.8	0.0
Deadlift	0.0	3.5	0.0	0.0	0.0	0.0	96.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Push Press	0.0	0.0	0.0	0.0	1.8	0.0	0.0	78.9	0.0	0.0	17.5	0.0	0.0	1.8	0.0
Back Squat	0.0	0.0	0.0	0.0	5.3	0.0	0.0	0.0	93.0	0.0	1.8	0.0	0.0	0.0	0.0
Arm Curl	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
Barbell Military Press	0.0	0.0	0.0	0.0	8.8	0.0	0.0	10.5	0.0	0.0	80.7	0.0	0.0	0.0	0.0
Barbell Bent–over Row	0.0	7.0	0.0	0.0	0.0	0.0	7.0	0.0	0.0	0.0	0.0	86.0	0.0	0.0	0.0
Burpee	0.0	14.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.0	78.9	0.0	0.0
Leg–raised Crunch	0.0	0.0	0.0	0.0	0.0	1.8	0.0	0.0	3.5	0.0	1.8	0.0	0.0	93.0	0.0
Lateral Raise	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.0	0.0	0.0	93.0

FIGURE 14. **Confusion matrix of workout classification. (row) True classes. (column) Inferred classes.**

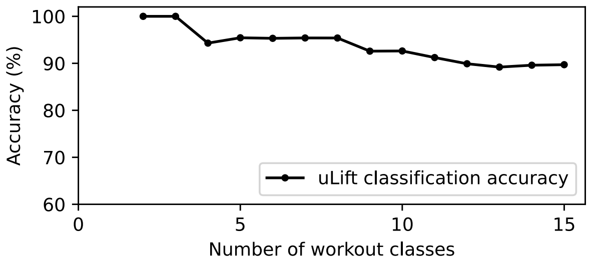


FIGURE 15. **Classification accuracy against number of workout classes.**

The results of the workout classification accuracy based on the number of templates are shown in Table 3. The classification accuracy remained stable (88-90%) regardless of the number of templates used, and the highest accuracy was achieved with three templates. Given this result, along with the fact that increasing the number of templates results in higher computational costs, the optimal number of templates was set to three. Additionally, the performance of uLift was compared to other fully automatic workout analysis algorithms, MiLift [19] and RecoFit [15], as well as deep learning methods based on convolutional neural networks [20] and recurrent neural networks [21]. For each of these algorithms, the number of sessions used for training varied from one to five, and the remaining sessions were used for testing. The result for each number of training sessions and the global averages are displayed in Table 3. Overall, uLift gave the best performance among these models, outperforming the other two fully automatic algorithms for all five numbers of sessions, and performed better than the deep learning models for session numbers up to three. As deep learning models require large amounts of data for training, it was expected that they become better with more sessions used for training. However, this characteristic of data demand makes deep learning models impractical to be deployed in a realistic environment. In addition, uLift, when evaluated on

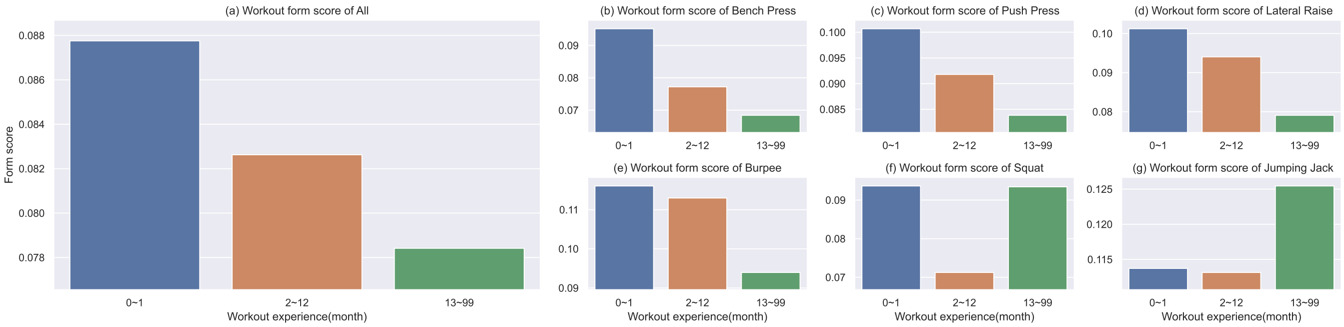


FIGURE 16. **Mean form scores by groups. (a) Mean workout form scores. (b)-(g) Mean form scores by the workout type.**

an Intel Core i9-10900X (20-cores) CPU, took the shortest inference time on average (24 ms) compared to RecoFit (36 ms), MiLift (37 ms), CNN (97 ms), and LSTM (719 ms). Figure 15 shows the classification accuracy of uLift against the number of workout classes. Starting from 100% with 2 and 3 classes, the accuracy slightly decreased to 90.06% with 15 classes, showing the potential scalability of uLift with an increasing number of workouts.

The confusion matrix that represents the result of the workout classification using three templates is shown in Figure 14. Lunge motion displayed the lowest accuracy in the confusion matrix. This low accuracy can be attributed to the asymmetric nature of lunge movements combined with the fact that they do not fundamentally involve hand movements, allowing participants to change hand positions midway through the exercise, which is a significant contributor to the decreased accuracy. In addition, push press movements showed relatively low accuracy and were mostly confused with the military press, a movement that shares many similarities with the push press. The principal difference between the two movements arises from whether the lower body momentum is utilized at the outset of the movement. Similarly, misclassification was observed in Burpee motion, primarily because it encompasses a push-up position at a certain stage, which can cause confusion in the classification. Given the variability in the workouts, the classification results showed a robust performance. Moreover, uLift demonstrates crucial adaptability by being able to immediately incorporate new workout types without the need for a training process.

D. WORKOUT QUALITY ASSESSMENT

Workout quality is suggested as a score that reflects the consistency of posture across each repetition of a workout. The form score was measured for three subgroups of participants divided by their workout experiences. The basic hypothesis was that workout experience distinguishes how consistently each participant performs workouts. Thus, the form score could serve as an indirect workout quality measurement.

Of the three subject subgroups, the first group was the beginner group, with the criterion of less than two months of workout experience. This group was expected to have little experience with workouts, thus exhibiting inconsistent movements over the repetitions. The second group comprised the intermediate group, in which the participants' workout experience ranged from 2 to 12 months. These participants had a certain level of knowledge about exercise performance and could understand most workout postures with little guidance. The last group consisted of participants with workout experience extending beyond one year (> 12 months). Members of this group were adept at performing most workouts independently. Furthermore, a simple verbal explanation is typically sufficient for them to understand and perform previously unknown workouts.

The form scores for the three groups are shown in Figure 16. These groups had significantly different form scores; the more experienced groups had lower form scores, indicating that they were more stable and consistent (Kruskal-Wallis test, $H(2) = 36.3$, $p = 1.29 \times 10^{-8}$). Post-hoc analysis using Dunn's test revealed that the most experienced group (> 12 months) had a significantly lower form score than all other groups. In addition, the second-experience group (2-12 months) had a significantly lower form score than the least-experience group (< 2 months). Significance was determined at the level of $\alpha = 0.05$ with Holm adjustment. However, there were exceptions; the form scores of squats, lunges, jumping jacks, and Deadlift were higher in the most experienced group. There are various reasons for this finding. One possible reason is that when a given workout was simple and basic, the more experienced participants tended to give some variation, while the less experienced participants simply tried to follow simple routines.

V. CONCLUSION

In this paper, we present uLift, an adaptive workout tracker that automatically detects the workout state, counts repetitions, classifies workout types, and seamlessly assesses workout quality without the need for user intervention. To evaluate our methods, we collected a multi-joint workout dataset based on an anaerobic workout in a real gymnasium environment from participants with various workout experiences. Compared to other deep learning methods, uLift's algorithms do not require a learning process, which makes it a practical and adaptable solution for workout tracking. Moreover, uLift effectively adapts to continuously changing workout patterns and possesses the capability to add or remove workout types without the need for retraining. Our evaluation results for workout detection, repetition counting, and workout classification exhibited an accuracy comparable to that of machine learning methods. In addition, the workout quality measured by the form score matched well with the three participant groups divided by their workout experiences. This is an encouraging result because uLift's assessment of workout quality does not depend on manual, possibly subjective, annotations from human experts. Finally, uLift's lightweight algorithms and minimal dependence on hardware configurations make it adaptable to widely different scenarios for workout tracking.

ACKNOWLEDGMENT

(Jongkuk Lim and Youngmin Oh contributed equally to this work.)

REFERENCES

[1] L. Vanhees, J. Lefevre, R. Philippaerts, M. Martens, W. Huygens, T. Troosters, and G. Beunen, "How to assess physical activity? How to assess physical fitness?" *Eur. J. Cardiovascular Prevention Rehabil.*, vol. 12, no. 2, pp. 102–114, Apr. 2005.

[2] A. Cordero, M. D. Masiá, and E. Galve, "Physical exercise and health," *Revista Española de Cardiología (English Edition)*, vol. 67, no. 9, pp. 748–753, Sep. 2014.

[3] C.-C. Yang and Y.-L. Hsu, "A review of accelerometry-based wearable motion detectors for physical activity monitoring," *Sensors*, vol. 10, no. 8, pp. 7772–7788, Aug. 2010.

[4] F. J. Penedo and J. R. Dahn, "Exercise and well-being: A review of mental and physical health benefits associated with physical activity," *Current Opinion Psychiatry*, vol. 18, no. 2, pp. 189–193, Mar. 2005.

[5] R. G. Laforge, J. S. Rossi, J. O. Prochaska, W. F. Velicer, D. A. Levesque, and C. A. McHorney, "Stage of regular exercise and health-related quality of life," *Preventive Med.*, vol. 28, no. 4, pp. 349–360, Apr. 1999.

[6] R. Belardinelli, F. Capestro, A. Misiani, P. Scipione, and D. Georgiou, "Moderate exercise training improves functional capacity, quality of life, and endothelium-dependent vasodilation in chronic heart failure patients with implantable cardioverter defibrillators and cardiac resynchronization therapy," *Eur. J. Preventive Cardiol.*, vol. 13, no. 5, pp. 818–825, Oct. 2006.

[7] A. E. Latimer-Cheung, L. A. Pilutti, A. L. Hicks, K. A. M. Ginis, A. M. Fenuta, K. A. MacKibbon, and R. W. Motl, "Effects of exercise training on fitness, mobility, fatigue, and health-related quality of life among adults with multiple sclerosis: A systematic review to inform guideline development," *Arch. Phys. Med. Rehabil.*, vol. 94, no. 9, pp. 1800–1828, Sep. 2013.

[8] *The Official Fitbit Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.fitbit.com/home

[9] *The Official Apple Watch Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.apple.com/watch/

[10] *The Official Samsung Galaxy Watch Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.samsung.com/global/galaxy/galaxy-watch6/

[11] *The Official Garmin Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.garmin.com/en-U.S./

[12] *The Official Strong Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.strong.app/

[13] *The Official Jefit Site*. Accessed: Oct. 13, 2023. [Online]. Available: https://www.jefit.com/

[14] M. Rippetoe and L. Kilgore, *Starting Strength*. Wichita Falls, TX, USA: Aasgaard Company, 2017.

[15] D. Morris, T. S. Saponas, A. Guillory, and I. Kelner, "RecoFit: Using a wearable sensor to find, recognize, and count repetitive exercises," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2014, pp. 3225–3234.

[16] H. Koskimäki and P. Siirtola, "Recognizing gym exercises using acceleration data from wearable sensors," in *Proc. IEEE Symp. Comput. Intell. Data Mining (CIDM)*, Dec. 2014, pp. 321–328.

[17] H. Koskimäki, P. Siirtola, and J. Röning, "MyoGym: Introducing an open gym data set for activity recognition collected using myo armband," in *Proc. UbiComp/ISWC*, 2017, pp. 537–546.

[18] A. Soro, G. Brunner, S. Tanner, and R. Wattenhofer, "Recognition and repetition counting for complex physical exercises with deep learning," *Sensors*, vol. 19, no. 3, p. 714, Feb. 2019.

[19] C. Shen, B.-J. Ho, and M. Srivastava, "MiLift: Efficient smartwatch-based workout tracking using automatic segmentation," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1609–1622, Jul. 2018.

[20] T. T. Um, V. Babakeshizadeh, and D. Kulić, "Exercise motion classification from large-scale wearable sensor data using convolutional neural networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 2385–2390.

[21] K. Xia, J. Huang, and H. Wang, "LSTM-CNN architecture for human activity recognition," *IEEE Access*, vol. 8, pp. 56855–56866, 2020.

[22] N. Hosein and S. Ghiasi, "Wearable sensor selection, motion representation and their effect on exercise classification," in *Proc. IEEE 1st Int. Conf. Connected Health: Appl., Syst. Eng. Technol. (CHASE)*, Jun. 2016, pp. 370–379.

[23] A. Ebert, M. Kiermeier, C. Marouane, and C. Linnhoff-Popien, "SensX: About sensing and assessment of complex human motion," in *Proc. IEEE 14th Int. Conf. Netw., Sens. Control (ICNSC)*, May 2017, pp. 327–332.

[24] A. Hussain, K. Zafar, A. R. Baig, R. Almakki, L. AlSuwaidan, and S. Khan, "Sensor-based gym physical exercise recognition: Data acquisition and experiments," *Sensors*, vol. 22, no. 7, p. 2489, Mar. 2022.

[25] X. Guo, J. Liu, and Y. Chen, "FitCoach: Virtual fitness coach empowered by wearable mobile devices," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, May 2017, pp. 1–9.

[26] Y. Kowsar, M. Moshtaghi, E. Velloso, C. Leckie, and L. Kulik, "An online unsupervised dynamic window method to track repeating patterns from sensor data," *IEEE Trans. Cybern.*, vol. 52, no. 6, pp. 5148–5160, Jun. 2022.

[27] E. Velloso, A. Bulling, H. Gellersen, W. Ugulino, and H. Fuks, "Qualitative activity recognition of weight lifting exercises," in *Proc. 4th Augmented Hum. Int. Conf.*, Mar. 2013, pp. 116–123.

[28] W. Coates and J. Wahlström, "LEAN: Real-time analysis of resistance training using wearable computing," *Sensors*, vol. 23, no. 10, p. 4602, May 2023.

[29] A. Spilz and M. Munz, "Automatic assessment of functional movement screening exercises with deep learning architectures," *Sensors*, vol. 23, no. 1, p. 5, Dec. 2022.

[30] I. Pernek, G. Kurillo, G. Stiglic, and R. Bajcsy, "Recognizing the intensity of strength training exercises with wearable sensors," *J. Biomed. Inform.*, vol. 58, pp. 145–155, Dec. 2015.

[31] Y. Kowsar, M. Moshtaghi, E. Velloso, L. Kulik, and C. Leckie, "Detecting unseen anomalies in weight training exercises," in *Proc. 28th Austral. Conf. Comput.-Hum. Interact.*, 2016, pp. 517–526.

[32] J. Margarito, R. Helaoui, A. M. Bianchi, F. Sartor, and A. G. Bonomi, "User-independent recognition of sports activities from a single Wrist-Worn accelerometer: A template-matching-based approach," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 4, pp. 788–796, Apr. 2016.

[33] B. Zhou, M. Sundholm, J. Cheng, H. Cruz, and P. Lukowicz, "Never skip leg day: A novel wearable approach to monitoring gym leg exercises," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2016, pp. 1–9.

[34] S. Ishii, A. Yokokubo, M. Luimula, and G. Lopez, "ExerSense: Physical exercise recognition and counting algorithm from wearables robust to positioning," *Sensors*, vol. 21, no. 1, p. 91, Dec. 2020.

[35] D. Džaja, M. Čibarić, G. Šeketa, and R. Magjarević, "Accelerometer-based algorithm for the segmentation and classification of repetitive human movements during workouts," *Automatika*, vol. 64, no. 2, pp. 211–224, 2023.

[36] *Smart Sensor: BNO055—Bosch*. Oct. 13, 2023. [Online]. Available: https://www.bosch-sensortec.com/bst/products/all_products/bno055

[37] *nRF51822—Nordic Semiconductor*. Oct. 13, 2023. [Online]. Available: https://www.nordicsemi.com/?sc_itemid=%7BE343E4D9-21F1-4FBC-881F-10320A687576%7D

[38] B. J. Mortazavi, M. Pourhomayoun, G. Alsheikh, N. Alshurafa, S. I. Lee, and M. Sarrafzadeh, "Determining the single best axis for exercise repetition recognition and counting on SmartWatches," in *Proc. 11th Int. Conf. Wearable Implant. Body Sensor Netw.*, Jun. 2014, pp. 33–38.

[39] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intell. Data Anal.*, vol. 11, no. 5, pp. 561–580, Oct. 2007.

JONGKUK LIM received the B.S., M.S., and Ph.D. degrees in computer engineering from Dankook University, Republic of Korea, in 2012, 2014, and 2020, respectively.

Since 2020, he has been the Chief Technology Officer with J.Marple Inc. His research interests include computer engineering and its applications in human-centric systems, including human motion analysis, fitness motion analysis, wearable computing for health monitoring, deep-learning techniques for biomechanics, and computer vision applications in sports analytics.

Dr. Lim received the first Minister's Award from the Ministry of Science and ICT of Korea at ITRC Global Makerthon, in 2017, and the Best Idea Award at ITRC Global Makerthon, in 2019.

YOUNGMIN OH (Member, IEEE) received the B.S. and M.S. degrees in physics education and physics from Seoul National University, Seoul, Republic of Korea, in 2002 and 2004, respectively, the M.S. degree in bio and brain engineering from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 2010, and the Ph.D. degree in neuroscience from the University of Southern California, Los Angeles, CA, USA, in 2015.

From 2016 to 2018, he was the Principal Engineer with Neofect Inc. He was a Staff Engineer with Samsung Electronics, between 2018 and 2020. Since 2021, he has been an Assistant Professor with the School of Computing, Gachon University, Seongnam, South Korea. He has authored seven articles and conferences. His research interests include human motor learning, human action recognition, stroke rehabilitation, sports science, and sign language recognition.

YOUNGGEUN CHOI received the B.S. degree in aerospace engineering and electrical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea, in 1998, and the M.S. and Ph.D. degrees in computer science from the University of Southern California, Los Angeles, CA, USA, in 2005 and 2010, respectively.

Since 2011, he has been a Faculty Member with the Computer Engineering Department, Dankook University, Yongin, Republic of Korea. His research interests include the development of wearable IoT systems, smart rehabilitation devices for stroke survivors, and deep-learning models for sign language recognition.