Workout Type Recognition and Repetition Counting with CNNs from 3D Acceleration Sensed on the Chest

Kacper Skawinski, Ferran Montraveta Roca, Rainhard Dieter Findling[0000−0002−6985−1298], and Stephan Sigg[0000−0001−6118−3355]

Department of Communications and Networking,
Aalto University, Maarintie 8, 02150 Espoo, Finland
{firstname.lastname}@aalto.fi

Abstract. Sports and workout activities have become important parts of modern life. Nowadays, many people track characteristics about their sport activities with their mobile devices, which feature inertial measurement unit (IMU) sensors. In this paper we present a methodology to detect and recognize workout, as well as to count repetitions done in a recognized type of workout, from a single 3D accelerometer worn at the chest. We consider four different types of workout (pushups, situps, squats and jumping jacks). Our technical approach to workout type recognition and repetition counting is based on machine learning with a convolutional neural network. Our evaluation utilizes data of 10 subjects, which wear a Movesense sensors on their chest during their workout. We thereby find that workouts are recognized correctly on average 89.9% of the time, and the workout repetition counting yields an average detection accuracy of 97.9% over all types of workout.

Keywords: Acceleration · Activity Recognition · CNN · Deep Learning · Movesense · Neural Networks · Workout · Sensors

1 Introduction

In recent years the ability to track sports and workout activities with has broadly become available with off-the-shelf mobile devices. Those devices, including fitness trackers, smart phones, and alike, usually feature positioning sensors such as GPS as well as IMU sensors, such as accelerometers and gyroscopes. Tracking characteristics for sports thereby range from tracking distance with positioning sensors to step detection and counting when walking [4, 5]. While extracting certain characteristics about a given sport by those sensors has previously been investigated, automatically extracting information for workout sessions with mixed workout types is more sparse [10]. It would be desirable for body-worn mobile devices to at first automatically recognize the type of workout done, and subsequently to analyze characteristics about the workout. Since nearly all mobile devices nowadays feature IMU sensors such as accelerometers, but due to their size and energy constraints not all devices feature position sensors such as

GPS, the automatic recognition should ideally work with IMU sensor data only. Towards this goal, one aspect that has not yet been investigated is the suitability of using a *single body-worn 3D accelerometer* for automatically detecting workout, recognizing the workout type, and counting repetitions in a given workout type.

In this paper we therefore present a methodology to automatically detect and recognize workout in mixed workout type sessions, and to count repetitions once a workout type has been determined. Our methodology uses a single 3D accelerometer only, worn at the chest. The contributions of this paper are:

- We propose a deep learning based methodology to recognize exercise being performed and to distinguish between four different exercise types: pushups, situps, squats, and jumping jacks. For this we utilize only a single 3D accelerometer worn at the chest.
- Based on the predicted workout, we utilize PCA and peak detection to count repetitions within different workout types.
- We record a data set containing the four exercise types with 10 subjects and in between 2-3 exercise recordings per subject, which contains a total of 55 workouts and a total of 583 workout repetitions. We evaluate our approach with this data.

2 Related Work

Wearable mobile devices with IMU sensors have become prevalent, since they enable a wide range of applications for their users. Examples for such devices include the Nike FuelBand and FitBit Flex [10], which can be used to give activity and workout feedback. An exemplary study for workout feedback is the RecoFit study [10]. RecoFit aims to give real-time and post-work feedback to sport trainees. The system bases its information extraction on data gathered with embedded IMU sensors with 50 Hz sampling rate. RecoFit encompasses three stages: automatically segmenting exercise periods, recognizing the exercise type, and counting the repetitions done in a given exercise. They at first smooth data with a Butterworth Lowpass filter (-60 dB damping at 20 Hz). Then a sliding window (width of 5 s, overlapping between windows of 96%) is applied to achieve windows of uniform length for subsequent processing. Based on this data they extract 24 features (auto correlation, RMS, mean, standart deviation, variance, integrated RMS, and frequency power bands). To distinguish between workout and no-workout, they use those windows and features to train a L2 linear support vector machine (L2-SVM). To subsequently differentiate between the 26 different types of possible workout, they train a multiclass SVM classifier. Once the workout type is defined, they apply principal component analysis (PCA) on only acceleration data to reduce it to one dimension, then employ a repetition counting algorithm on that data. In their evaluation they use data of 114 participants and 146 sport sessions. Their results indicate precision and recall bigger than 95% in the detection phase. For exercise recognition, they used circuits of 4, 7, and 13 exercises and achieved an accuracy of 99%, 98%, and 96% respectively. The counting with ± 1 accuracy reached a precision of 93%.

Javed [7] proposed a method for arm and elbow workout exercise recognition. They use the accelerometers embedded in a Samsung Galaxy S4 smartphone. The exercises they investigated were arm based, that means workouts such as Bicep Curl, Active Pronator, Active Supinator, Assisted Biceps, Isometric Biceps, and triceps workout. Their data recording uses raw accelerometer data, then apply a class conditional probabilities filter. The filtered data is classified with different classifiers from the Waikato Environment for Knowledge Analysis (WEKA) toolkit [13]. Their findings indicate that Random Forest and LMT classifiers yield better performance for their setup, with an accuracy of 99,5% and 99,83%, respectively.

Another study [3] also used accelerometers embedded in Android smartphones, but focused on activities such as walking (fast, slow, upstairs, downstairs), running, and aerobic dancing. They utilized mobile phones in two different postures: in-hand phone position and in-pocket phone position. They applied a digital low pass filter to separate the gravity component. Subsequently, data is handed to a robust supervised classifier. For their evaluation, they tested multiple classifiers and found Multilayer Perceptron (MLP) and SVM to yield best accuracies with 89.48% and 88.76% in the in-hand case, and MLP and RF to yield best accuracies with 89.72% and 72.27% in the in-pocket case. Finally, the combination of multiple classifier with fusion was evaluated. Accuracy thereby was enhanced to 91.15% with combining MLP, LogitBoost, and SVM for the in-hand case, and to 90.34% for in-pocket with combining MLP, RF, and SimpleLogistic.

In another study [6], several machine learning models, including KNN, SVM, and RF were utilized to classify transportation ways such as driving a car, riding a bicycle, riding a bus, walking, and running. The study utilizes data from accelerometers, gyroscope, and rotation sensors. To achieve uniform sampling rates across those sensors, upsampling with linear interpolation to a uniform sampling rate of 100 Hz was used, similar to [9]. A sliding window with length 1 s was applied to achieve uniform sample sizes. The best results were generally obtained with RF (overall accuracy 95,1%), although SVM (overall accuracy 94,41%) was better for walking and running.

Liang and Wang [8] utilized Convolutional Neural Networks (CNN) to enhance the accuracy of classification of transportation modes over traditional machine learning methods. In their approach, a sliding window is applied to acceleration sensor data recorded with 50 Hz using smartphones. Subsequently each window is processed by a CNN to differentiate between seven classes: stationary, walking, cycling, driving, taking a bus, subway or train. The data is smoothed with a Savitzky-Golay filter to reduce noise in mobile phone movements. From the 3D acceleration values they then computed the magnitude. In the sliding window they extracted a 512 value long window with window overlapping by 12.5%. Those windows are used as input for the CNN model, which after convolutional and pooling layers uses a fully connected layer to predict the transportation type target. Their system was able to obtain an accuracy of 94.48%.

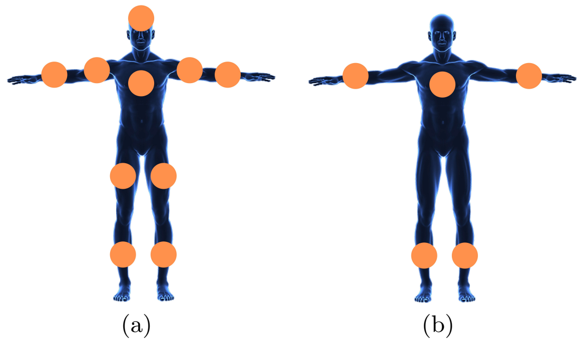


Fig. 1. Position of sensors for (a) most data gathering driven solution and (b) the setup evaluated to be best in literature [2, 12]. Figure adapted from [1].

3 Method

Our goal is to automatically detect, recognize, and count repetitions in four different types of workout. In this section we describe the workout types, data source and sensor position, as well as technical details of the detection, recognition, and repetition counting.

3.1 Workout Types and Sensor Position

The workout types we consider are four-fold:

- Pushups
- Situps
- Squats
- Jumping jacks

As our approach automatically detects if one of those types of workout is done, we also consider a fifth type, named no-workout, which covers all other types of workout or no workout being done at all.

Our goal is to recognize workout from 3D accelerometer data of a single sensor only. The reason for using only one sensor is usability: using more than one sensor would be cumbersome for users who want their workout to automatically be recognized and counted. In a preliminary study we evaluated five different sensor positions on the human body that have been shown to be useful for human activity recognition in previous studies [2, 12] (Fig. 1). This preliminary study yielded the chest to be the sensor position best suited to detect, recognize, and count repetitions from 3D acceleration sensor data only. For this reason we use an accelerometer on the chest in our approach.

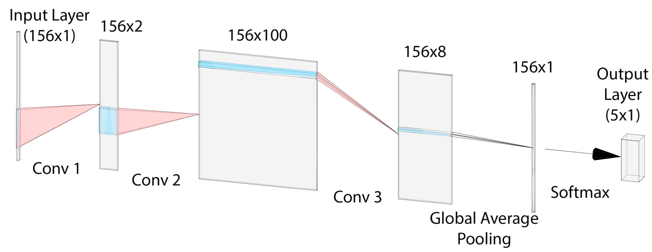


Fig. 2. The CNN architecture utilized for recognizing workout types.

3.2 Sensing Workout

We sense workouts with one 52 Hz 3D accelerometer on the chest. We use this sampling frequency due to related work with similar goals and good results utilizing similar frequencies [8, 10]. On the continuous 3D time series we apply a sliding window to split the stream into fixed length samples. Our window length is 1 s, and windows overlap for $\frac{1}{4}$ of their lengths, which corresponds to a $\frac{13}{52}$ s overlap. Each window position thereby yields a sample that consists of $52 \cdot 3 = 156$ features.

3.3 Workout Detection and Recognition

From labeled samples, each having 156 features and representing 1 s of sensed workout, we train a model to distinguish between our five workout types (the four target workout types and the no-workout type). This model will be able to automatically distinguish for new samples if workout is done, and if yes, which type of workout it is. The chosen model type is a convolutional neural network (CNN) with 3 hidden layers (Fig. 2). The network thereby has an input layer (156 neurons), three hidden fully connected convolutional layers, and a Softmax classification output layer (5 neurons) after GlobalAveragePooling1D. The first convolution is performed with 2 filters and kernel size 15, the second one with 100 filters and size 10, and the third one features 8 filers with size 2.

3.4 Workout Repetition Counting

After the type of workout has been determined for a given workout recording, our approach automatically detects the amount of repetitions in this workout. In contrast to workout type recognition this requires a longer window. We therefore only count repetitions once the workout type recognition yields that workout for a certain workout type has finished (with either another or no workout being started afterwards). With the data of one such continuous workout that contains 3D accelerometer data, we at first apply PCA and extract only the strongest PC dimension. This transforms the 3D time series into a 1D time series. To the resulting time series we then apply a peak detection to detect repetitions.

Table 1. Peak detection parameters for detecting repetitions in workout sessions.

Workout	d_{min}	h_{min}
Pushups	$\frac{15}{52}$ s	0.5
Situps	$\frac{15}{52}$ s	0.5
Squats	$\frac{15}{52}$ s	0.5
Jumping jacks	$\frac{5}{52}$ s	0.2

(a) Movesense sensor (b) Sensor worn on the chest

Fig. 3. (a) The utilized Movesense sensor with a 2 EUR coin for size comparison, and (b) the sensor being worn on the chest during workout.

The peak detection has two parameters: the minimum distance d_{min} between peaks, as well as the minimum height h_{min} of the peak. For the latter we set an adaptive threshold α (Eq. 1). Once we surpass α, the distance between this new candidate and the previous peak is calculated. If said distance is bigger than d_{min}, the candidate is counted as a peak.

$$\alpha = mean(\text{data}) + (max(\text{data}) - mean(\text{data})) \cdot h_{min} \tag{1}$$

We utilize different configuration of our peak detection for different workout types (Tab. 1). The amount of peaks corresponds to the amount of repetitions done in the workout, with exception of jumping jacks, for which the amount of repetitions is half the amount of peaks detected.

4 Evaluation

4.1 Utilized Sensor

For our study we utilize accelerometer values from a Movesense sensor [11] (Fig. 3). We selected the Movesense sensor due to its relatively compact hardware, with diameter of 36.6 mm and the thickness of 10.6 mm. The controlling unit is Nordic Semiconductor nRF52832 comprising a 32 –bit ARM Cortex-M4 with 64kB on-chip RAM and 512kB on-chip FLASH. The communication of the sensor is based on Low Energy Bluetooth 4.0. Movesense is able to measure linear and angular acceleration, magnetic field intensity, temperature, heart rate and ECG. The sensors for this research were provided by the sensor manufacturer,

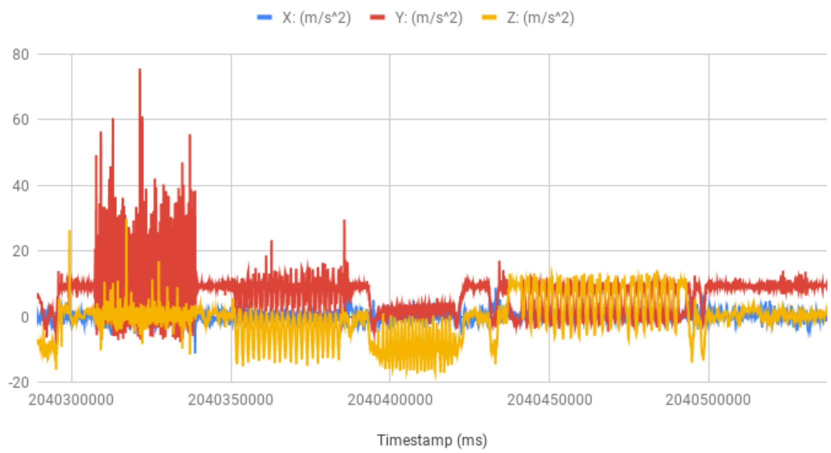
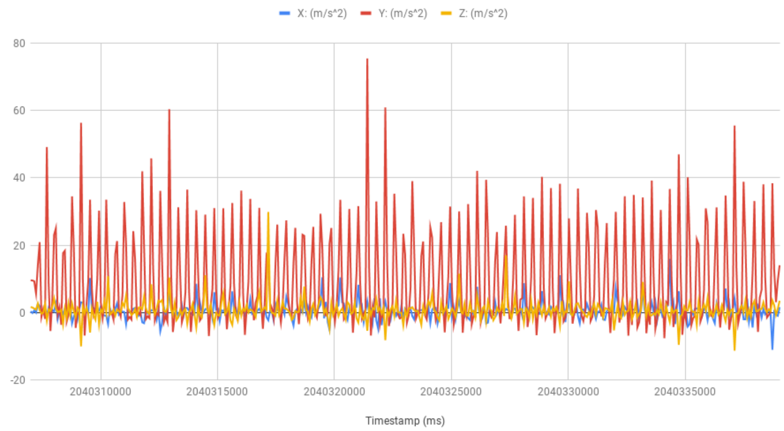


Fig. 4. Acceleration $[\frac{m}{s^2}]$ for an exercise set recorded on the chest. Contained exercises, from the left: jumping jacks, squats, pushups, and situps.

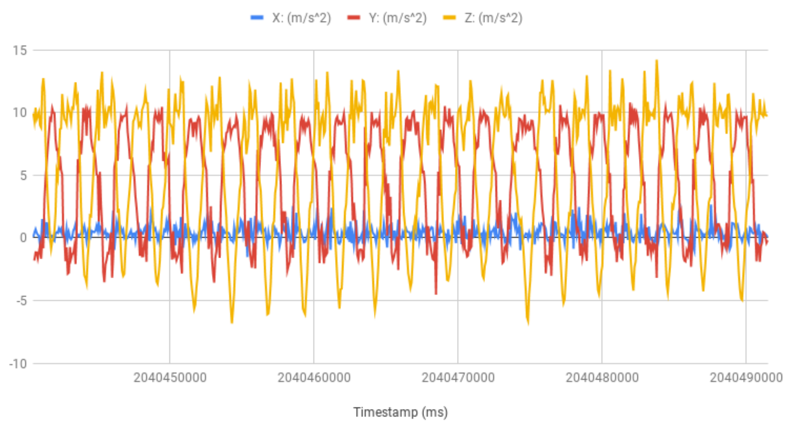
MoveSensor, which is part of the Suunto coorporation. As our study aims to use accelerometer data only, we utilize only the accelerometer embedded in the movesense sensor. This accelerometer is a 3D accelerometer and provides a sampling frequency of 12.5/26/52/104/208 Hz. For our evaluation we configured the sensor to sample with 52 Hz as a trade-of between energy sampling accuracy and energy consumption, similar to sampling frequencies used in related work for similar purposes [8, 10]. Participants wear the sensor on their chest using an attachment strap which is part of Movesense Developer Kit.

4.2 Evaluation Data

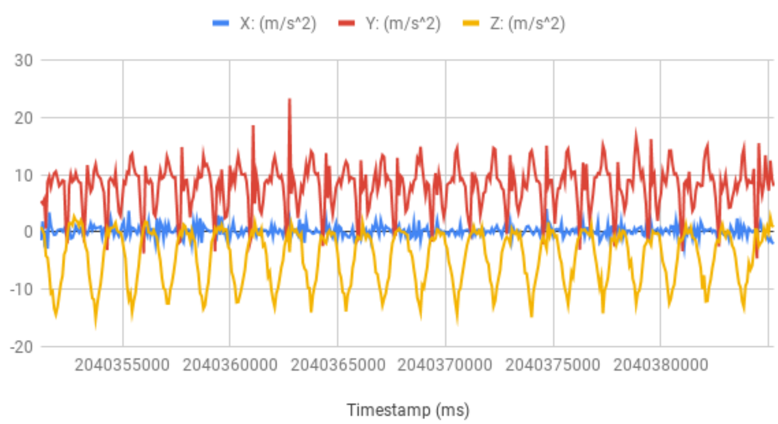
We recorded workout data of 10 different subjects with 2-3 workout sessions per subject, each performing the four workout types, and with no-workout phases in between workouts. Thereby, 11 workouts per workout type were recorded, resulting in a total of 44 workouts. Each workout thereby contains 10, 20 or 40 repetitions of the workout, depending on the workout type and the person. An example workout recording that contains jumping jacks, squats, pushups, and situps, in this order, is shown in Fig. 4. The no-workout parts of those samples contain diverse resting related activities (uncontrolled and individual for each participant), including sitting, standing, walking, drinking water, and the transition from one workout type to another one, like getting from push-ups. After data was recorded, the periods corresponding to the four target workout types, as well as all no-workout periods were annotated in the data. This resulted in a total of 55 workout samples (11 being no-workout). Examples for workout containing jumping jacks, situps, and squats, are shown in Fig. 5. Those extracted samples thereby form the basis for training and evaluating the workout type recognition and repetition counting.



(a) Jumping jacks

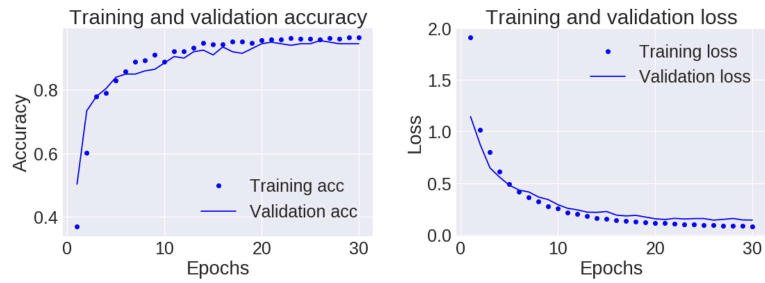


(b) Situps



(c) Squats

Fig. 5. Acceleration $[\frac{m}{s^2}]$ for samples with different workout types recorded on the chest.



(a) Accuracy (b) Loss

Fig. 6. Accuracy (a) and loss (b) over training and validation data.

4.3 Evaluation Setup

To train and evaluate our approach, we utilize a training-validation test set data partitioning approach on the recorded evaluation data set. The training, validation, and test set contain 75%, 17.5%, and 7.5% of all samples, respectively. The CNN model is trained with the training data and a batch size of 512. The validation data is used to monitor the training progress and to stop training when the accuracy over the validation set does not improve anymore, within a small tolerance, for four consecutive epochs. The test data is used to report the final workout detection and type recognition rates once the model has been trained.

5 Results

5.1 Exercise Type Recognition Results

Results for workout type recognition in general indicate good recognition results. The overall accuracy is 90.6% for the validation set and 89.9% for the test set, with a final loss over test data of 0.206 (Fig 6). The confusion matrix (Fig. 7) indicates only minor confusion between certain types of workout/no workout. Confusion is most frequent between no pushups and no exercise (10% of no pushups recognized as no exercise, 4% of no exercise recognized as pushups), and between situps and squats (10% of situps recognized as squats, and 7% of squats recognized as situps).

With the ready trained model, the time required to predict the workout type for one 1 s long workout sample was measured to be on average 39.74 μs, with a standard deviation of 6.79 μs (1000 predictions from test set samples). Those measurements were done on a Lenovo ThinkPad X1 Carbon with Intel Core i7-8550U 1.80GHz$\times$8 processor and 16 GB of memory.

5.2 Workout Repetition Counting Results

Once the workout type has been recognized, the workout repetition counting yields accuracies in between 97.4%-98.7% (Tab. 2). While errors with pushups

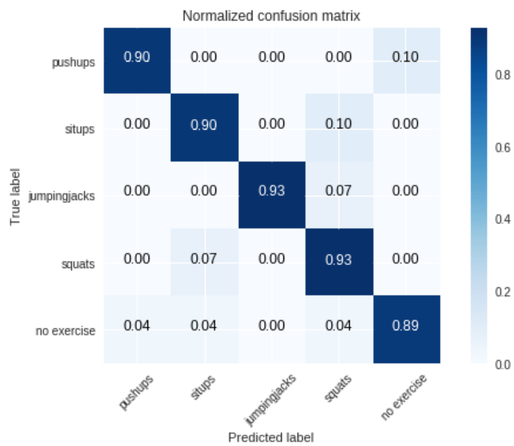


Fig. 7. Normalized confusion matrix.

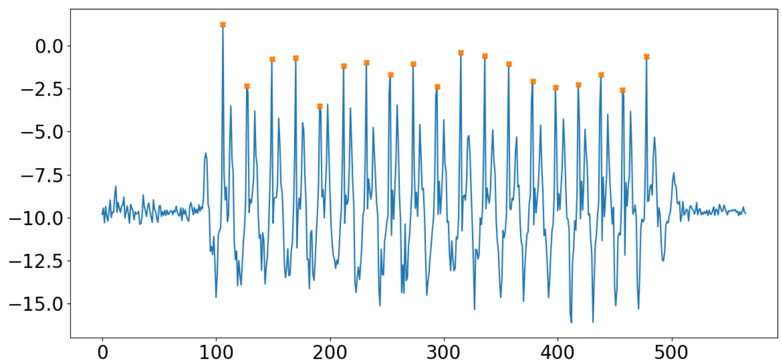
Table 2. Repetition counting results per type of workout, with the total contained repetition, the repetitions detected by our approach, the amount of false positives and false negatives, and the accuracy resulting thereof.

Workout	Contained	Detected	False negatives	False positives	Accuracy
Pushups	116	113	3	0	97.4%
Situps	159	161	0	2	98.7%
Squats	136	139	0	3	97.8%
Jumping jacks	172	168	4	0	97.7%

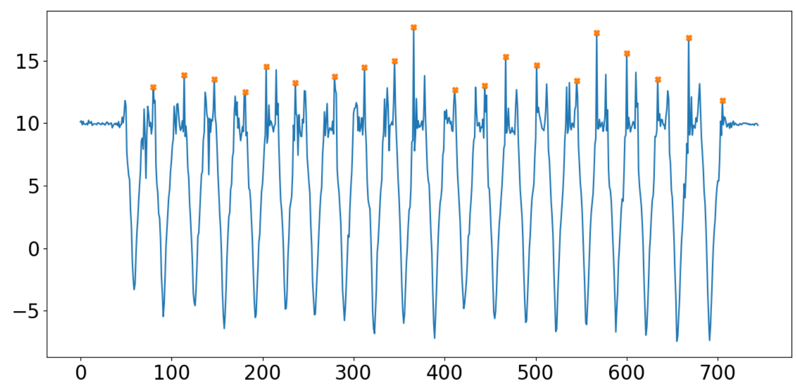
and jumping jacks were caused by false negatives (repetitions not being detected), errors with situps and squats were caused by false positives (falsely detecting a repetition where there is none). The average detection accuracy of repetitions over the four workout types thereby is 97.9%. Examples for repetitions being detected in a continuous workout for doing pushups, situps, and jumping jacks are shown in Fig. 8.

6 Conclusions

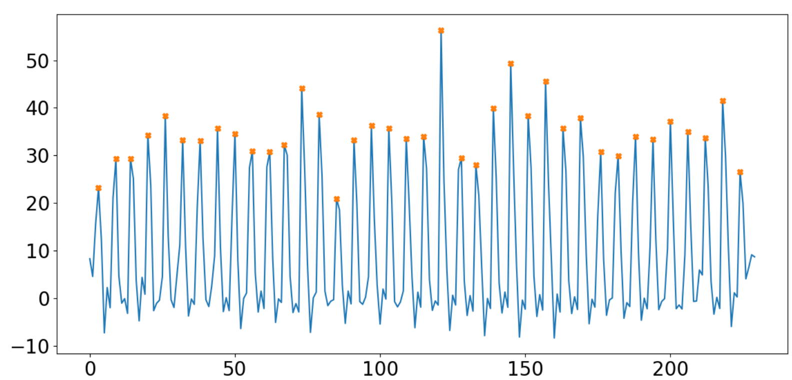
This paper has presented a methodology to recognize four different workout types and count workout repetitions from 3D acceleration sensor data of the chest area. Our approach at first detects the type of workout, or, if no workout is performed, using a 5 layered CNN model. Once the workout types has been determined we utilize a PCA and peak detection based algorithm to count the repetitions inside a workout session of one workout type. For evaluating our approach we utilize a self-recorded data set of 10 subjects with a total of 55 continuous workout periods and a total of 583 workout repetitions. Results indicate our workout type recognition to detect the workout type, or no workout being performed,



(a) Pushups



(b) Situps



(c) Jumping jacks

Fig. 8. Peak counting based on the acceleration $[\frac{m}{s^2}]$ sensed for samples with (a) pushups, (b) situps, and (c) jumping jacks.

correctly with an average accuracy of 89.9%. Once the workout type has been determined by our approach, results for workout repetition counting indicate an average counting accuracy of 97.9%. One limitation of our study is the limited insight into the suitability of different sensor positions on the human body for detecting those workout types and counting repetitions in them. Future work could therefore investigate and compare the suitability of different positions on the human body to wear the sensor on.

References

1. Archerydirect.co.nz.: Human body figure (2019), `http://www.archerydirect.co.nz/images/assetimages/human.png`
2. Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L., Amirat, Y.: Physical human activity recognition using wearable sensors. MDPI Sensors **15**, 31314–31338 (2015)
3. Bayat, A., Pomplun, M., Tran, D.A.: A study on human activity recognition using accelerometer data from smartphone. The 11th international conference on mobile systems and pervasive computing pp. 1–8 (2014)
4. Bergman, C., Oksanen, J.: Conflation of openstreetmap and mobile sports tracking data for automatic bicycle routing. Transactions in GIS pp. 848–868 (2016)
5. Crema, C., Depari, A., Flammini, A., Sisinni, E., Haslwanter, T., Salzmann, S.: Imu-based solution for automatic detection and classification of exercises in the fitness scenario. In: 2017 IEEE Sensors Applications Symposium (SAS). pp. 1–6 (March 2017). https://doi.org/10.1109/SAS.2017.7894068
6. Jahangiri, A., Rakha, H.A.: Applying machine learning techniques to transportation mode recognition using mobile phone sensor data. Strategic management journal **16**(5), 2406–2417 (2015)
7. Javed, T., Awan, M.A., Hussain, T.: Recognition of arm & elbow exercises using smartphone's accelerometer. NFC-IEFR Journal of Engineering & Scientific Research pp. 1–6 (2017)
8. Liang, X., Wang, G.: A convolutional neural network for transportation mode detection based on smartphone platform. Strategic management journal **18**(7), 338–342 (2017)
9. Manzoni, V.: Transportation mode identification and real-time CO2 emission estimation using smartphones. Strategic management journal (2010)
10. Morris, D., Saponas, T.S., Guillory, A., Kelner, I.: RecoFit: Using a wearable sensor to find, recognize, and count repetitive exercises. Strategic management journal pp. 3225–3234 (2014)
11. Movesense: Movesense sensor description (2019), `https://www.movesense.com/product/movesense-sensor/`
12. Patel, S., Park, H., Bonato, P., Chan, L., Rodgers, M.: A review of wearable sensors and systems with application in rehabilitation. JNER **9**(21) (2012)
13. Weka: Waikato environment for knowledge engineering. New Zealand Computer Science Research Students Conference, University of Waikato, Hamilton, New Zealand pp. 57–64 (1995)