Smart-Mat: Recognizing and Counting Gym Exercises with Low-cost Resistive Pressure Sensing Matrix

Mathias Sundholm
German Research Center for Artificial Intelligence (DFKI)
Trippstadter Straße 122
67663, Kaiserslautern, Germany
mathias.sundholm@dfki.de

Jingyuan Cheng
German Research Center for Artificial Intelligence (DFKI)
Trippstadter Straße 122
67663, Kaiserslautern, Germany
jingyuan.cheng@dfki.de

Bo Zhou
German Research Center for Artificial Intelligence (DFKI)
Trippstadter Straße 122
67663, Kaiserslautern, Germany
bo.zhou@dfki.de

Akash Sethi
German Research Center for Artificial Intelligence (DFKI)
Trippstadter Straße 122
67663, Kaiserslautern, Germany
akash.sethi@dfki.de

Paul Lukowicz
DFKI and University of Kaiserslautern
Trippstadter Straße 122
67663, Kaiserslautern, Germany
paul.lukowicz@dfki.de

ABSTRACT
There is a large class of routine physical exercises that are performed on the ground, often on dedicated "mats" (e.g. push-ups, crunches, bridge). Such exercises involve coordinated motions of different body parts and are difficult to recognize with a single body worn motion sensors (like a step counter). Instead a network of sensors on different body parts would be needed, which is not always practicable. As an alternative we describe a cheap, simple textile pressure sensor matrix, that can be unobtrusively integrated into exercise mats to recognize and count such exercises. We evaluate the system on a set of 10 standard exercises. In an experiment with 7 subjects, each repeating each exercise 20 times, we achieve a user independent recognition rate of 82.5% and a user independent counting accuracy of 89.9%. The paper describes the sensor system, the recognition methods and the experimental results.

Author Keywords
Sport tracking; activity recognition; pressure sensors; health

ACM Classification Keywords
I.2.1. Artificial Intelligence: Applications and Expert Systems

UbiComp 14, September 13–17, 2014, Seattle, WA, US.

ACM 978-1-4503-2968-2/14/09...$15.00.
http://dx.doi.org/10.1145/2632048.2636088

INTRODUCTION
Motivating people to exercise and helping them monitor their personal progress is a popular and well researched application of ubiquitous computing. However the majority of systems (including many commercial offerings) developed to date are designed for tracking aerobic training such as running and cycling. Such exercises are mostly defined by speed, distance and terrain which can be reliably monitored using a single unobtrusive on-body motion sensor and a GPS receiver contained for example in a smart phone. Much less work, particularly in terms of broadly usable unobtrusive systems, exists in the area of strength and muscular endurance training. Such training increases bone strength and muscular fitness, and reduces risk for injuries [1]. We distinguish two types of strength oriented exercises. The first is based on special machines typically found in fitness studios. The second involves exercises such as push-ups, crunches or free lifting weights which can be performed anywhere. In this paper we focus on monitoring the latter. Since the exercises are defined by often complex motion patterns of different body parts they, in general, require more elaborate sensor systems than a single body mounted motion sensor.

This paper is motivated by the observation that many strength related exercises involve contact between different body parts and the ground. For each exercise a characteristic spatio temporal contact and pressure pattern can be identified. We thus propose to use a resistive, textile pressure sensor matrix for tracking exercises such as push-ups, abdominal crunches and squats. Such a matrix can be easily integrated into standard gym-mats often used in this context. This means that the monitoring can be performed without the need to wear any additional sensors by merely exchanging a normal gym-mat for a smart one.

The main contributions of this paper are:

- We introduce a resistive pressure mat as a new sensing modality for ubiquitously recognizing and counting exercises. The mat is thin, soft and low-cost and can easily replace traditional gym mats.

- We show that our pressure sensor mat successfully distinguishes 10 common gym exercises. Using a kNN-classifier we achieve an overall accuracy of 88.7% for 7 subjects individually and an 82.5% person independent recognition rate. By majority voting we can classify a set of exercises with 86.5% accuracy, and with the correct class being in the top 2 of the classifier output 94.3% of the time.

- We demonstrate that the system can robustly count the number of performed exercises within real-life data. By utilizing dynamic time warping and a method for selecting the best matching template from a training set collected from several users performing the exercises at different speeds we achieve an overall 89.9% person independent detection rate for 10 activities, and 7 subjects.

- We investigate in detail the impact of variation in the execution speed of exercises, and the influence of testing and training the system on different users.

The article starts by presenting the state of the art in sport tracking and activity recognition. We then present our pressure sensor platform used in the experiments. We describe the experiment and the taxonomy of exercises. The evaluation is split into two main parts. In the first part we evaluate how exercises are distinguished by calculating rotation and translation invariant features from the pressure data and using a kNN-classifer for classification. In the second part we describe how the exercises can be robustly detected and counted by utilizing dynamic time warping. Finally we state our conclusions and future work.

RELATED WORK

As a method for motivating people to exercise regularly, different kinds of sport tracking systems have been proposed. A study shows that, users who get computer feedback during their exercise sessions have significantly higher attendance and adherence and also fewer drop-outs compared to users without any feedback [7].

Sport tracking systems for tracking aerobic exercises are both common and commercially available [5, 2]. These systems use GPS, heart-rate monitors and/or pedometer to track the distance a person is walking, running or cycling, and to estimate the burned calories. Systems for tracking strength training is far less common.

FitLinxx proposes a strength training tracking system that is mounted to weight-lifting machines in the gym [3]. The user can log on to their personal training profiles using touch screens. The system tracks the amount of weight the user is using, the number repetitions performed, and give users direct feedback of their progress. The system requires the gym to be equipped with special exercise machines and infrastructure, which is very expensive and not suitable for average user.

A different approach to track strength training has been to use on-body sensors. Chang et al. uses accelerometers positioned on users hands and waist to track free-weight training [8]. The system is able to recognize different gym exercises and also to count the number of repetitions. Seeger et al. improves on this work by recognizing exercises in real-time through out the day, using a heart-rate monitor and accelerometers attached to upper arm and leg [16].

Other systems exploit only the inertial sensors in smartphones to detect and count sport exercises. The smartphone has gained a lot of attention in activity recognition, since it can through its sensors capture user's motion, analyse it in real-time, provide instant user feedback, and log data for long-term analysis. Muehlbauer et al. investigated how a holster worn smartphone can recognize different upper body exercises and count repetitions [13]. Pernek et al. build upon this work by supporting a broader range of exercises including some gym machines [14].

Compared to previous work we introduce an entirely different approach to ubiquitously recognize and count exercises, by using a pressure sensor mat. In our approach the user do not have to wear any sensors or devices that might be distracting while performing exercises. The user can perform exercises in the same way as on a traditional gym mat, and the system can automatically detect what exercises are performed and how many repetitions are done.

BASIC IDEAS

Hardware Platform

The core of the gym mat is a resistive pressure sensor matrix. The matrix is built out of a thin layer of conductive polymer fiber sheet. The conductive sheet is positioned between 80 parallel stripes of conductive foil on each side, spaced 1 cm between each other, resulting in a 80 cm $\times$ 80 cm sensor mat.

The volume resistance of the fiber sheet/film reduces locally when the material is pressed, which can be measured by the crossing foil electrodes. The sampling electronics is a custom printed circuit board, which follows our study on the multi-channel parallel architecture [17]. It has one FPGA to control the system, and one FPGA and ultra-fast analog switches to sweep through every column. Three dedicated 24-bit ADCs with analog routing circuitry samples one column of the matrix at a time, creating a 80 $\times$ 80 pixel frame every 1/40 second. The data is sent to the computer through 3 serial ports (UART-USB) by a dedicated bridge chip through one physical USB cable.

The result is a thin, soft and low-cost sensor mat, that in the future could replace traditional gym mats. Pictures of our current prototype can be seen in Figure 1.

Preprocessing of Frames

The data frames produced by the sensor mat are calibrated by removing DC and noise. The DC component for each sensor element in the matrix is calculated as the average value during the time the sport mat is not occupied by a person. When a force is applied on the mat the voltage in that position is increasing. However due to electrical properties of the sensor

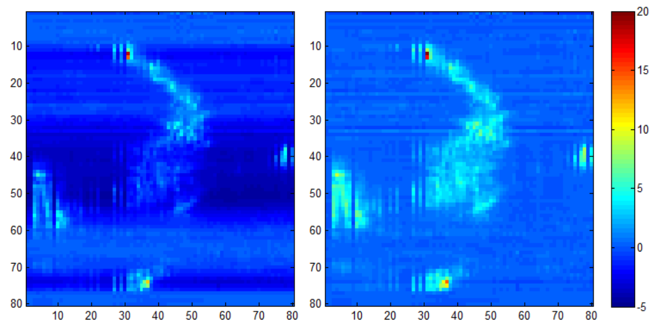


Figure 2: Frame before and after preprocessing.

the entire row is affected and slightly pulled down as can be seen in Figure 2 (a). To compensate for this, a compensation term is calculated for every row of the frame as the minimum value of the row. For robustness the frame is filtered by a 3 × 3 median filter before calculating the minimum. The resulting preprocessed frame after calibration and row compensation is shown in Figure 2 (b).

Signals and Recognition Approach

Examples of signals for different types of exercises are shown in Figure 3. With the spatial resolution of around 1 cm, the mat can distinguish the body parts that are in contact with the ground. We can also see changes in weight distribution both between body parts and within a body part. With a sampling rate of 40 Hz we are also able to observe the temporal evolution of the weight distribution. Thus the information that we can use for monitoring the exercises is based on (1) the body parts that are on the mat (hand, leg, knee, buttock, back), (2) the relative position and orientation of those body parts, (3) the sequence of the (1) and (2), (4) the weight distribution within and between body parts and (5) the temporal evolution of this sequence. The weight distribution contains important information, not only about the body parts which have contact with the mat, but also about other parts of the body. Thus, for example in Figure 3 II the weight distribution between (and within) the hand and the knee which are on the mat is strongly influenced by the motion of the leg and the hand which are in the air. Similarly balance changes related weights being lifted in Figure 3 IX translate into balance changes in the feet which are registered by the mat.

From the above considerations, our approach consist of (1) signal conditioning, (2) detecting areas on which there is body part contact, (3) computing a set of features that represent the information sketched above in a compact, abstract way and (4) performing a classification on those features. Building on the classification we then perform the counting of the number of repetitions of each exercise.

IDENTIFICATION OF ACTIVITIES

Some activities cause changes in the weight and pressure while some activities cause changes in the shape and size of the contact area. Therefore we calculate features that describe the change in weight distribution and the contact area.

The classification process can be summarized by the following steps:

1. Calculate area A, weight W, pressure p and Hu's Seven Moments ϕ_{1-7} for each frame in the recording. All activities on the carpet are described by these 10 time series.
2. For each labelled activity, statistical features are calculated for describing the variation of the 10 time series during the exercises.
3. Classification is performed using a kNN-classifier.

Calculation of Invariant Features

While the user is performing exercises on the mat the weight distribution, contact area and shape of the contact area is changing. To calculate the size of the contact area for an individual frame in the recording we threshold the pressure image. The area A is the the sum of all pixels above the threshold. Before thresholding we upsample the image to 160 × 160 pixels; intermediate pixels are interpolated using bi-cubic interpolation scheme. This creates a higher resolution frame that is advantageous when calculating the area, the weight of the frame. The weight W is calculated as the sum of all pixels within the area. We also define pressure as $p = \frac{W}{A}$. Pressure gives information of how concentrated the force is. For example when the user is standing the area is small and the pressure is large. When the user is lying the pressure is lower because of the larger contact area.

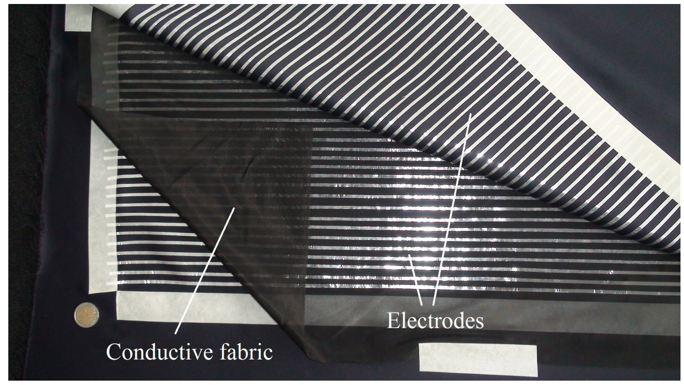
To compactly describe the shape of the contact area and weight distribution of the image we use image moments. Image moments are scalar quantities that statistically describe the intensity distribution of the image and have been since long used in the pattern recognition community for image and shape recognition [10] [12]. Hu introduced seven image moments that are invariant to translation, scale and rotation [10]. While Hu's moments are most often used to classify shapes of binary images they are not limited to only binary images. To describe pressure distribution we calculate Hu's seven moments ϕ_{1-7} for each pressure frame of the recording. Before calculating the moments the image is normalized so that the total weight is 1, which removes the weight information already included in W.

By calculating these features for each frame we reduce the dimensionality of our original data to a set of 10 time series $\mathbb{F} = \{A(t), W(t), p(t), \phi_{1-7}(t)\}$ denoted as feature domain. Because each of these features are translation and rotation invariant it does not matter where the user stands/lies on the sensor mat or in which direction he/she is facing while performing the exercises. The classification and counting of exercises is done in feature domain $\mathbb{F}$.

Experimental Setup

To evaluate our approach, 7 healthy test subjects (age 23-27) are asked to perform 10 predefined sport exercises on the sensor carpet. The exercises are grouped into sets. Each exercise is repeated 10 times per set and each set is performed 2 times (total 10 exercises × 10 repeats × 2 sets). To also evaluate how well our system is able to detect exercises performed at

(a)



(b)

Figure 1: (a) Top view of our 80 × 80 cm sensor matrix prototype. (b) The core of the sensor is a conductive textile positioned between stripes of electrodes.

different tempo we ask the subjects to perform the first set fast and the second set in a more relaxed tempo. As a result our test subjects performed the first set 1.5 times faster compared to the second set on average. The exercises are video recorded and the starting point and end point of each exercise is labelled manually based on the recorded video.

The chosen sport activities are standard exercises that can be performed on a normal gym mat and selected from a pool of exercises that are recommended by experts at Mayo Clinic [4]. Since we are interested in investigating whether the sensor carpet is able to also count the exercises, all the selected exercises are activities that are performed in repeats. For chest press and biceps curl we used 2.5–5.0 kg dumbbells. The exercises performed are shown in Figure 3.

Classification and Results

To classify the exercises we rely on the temporal variation of time-series in feature domain $\mathbb{F}$ during the exercise. The following features are calculated for each time-series in $\mathbb{F}$:

- mean and standard deviation
- maximum, minimum, and maximum - minimum
- maximum and minimum of the approximate 1st derivative

Each feature is normalized by the standard deviation of the corresponding feature in the training set before classification.

When we classify the exercises of each user using 10-fold cross validation with an kNN-classifier we reach almost 95-100 % classification rate for each subject. Average confusion matrix shown in Figure 4 (a). This result might however be too optimistic. The exercises within one set are fairly similar, while there are larger variation (in speed, position) between exercises performed in different sets. Therefore we instead perform a repeated hold out where the exercises is split so that the one set of exercises is used for training and the other for testing. Using this method we reach 88.7 % average accuracy for 7 subjects. The confusion matrix is shown in Figure 4 (b).

To evaluate the impact of person independent training we perform a leave-one-out cross validation, where one subject is left out for testing while the other subjects are used for training. We reach an average accuracy of 82.5% for person independent classification across 7 subjects. The confusion matrix is shown in Figure 4 (c). This suggests that our system works well across different users. The most missclassification happens between calf raise and squats. Both of these exercises are performed standing. It also appears that some users are lifting their heels while performing squats, making the change of contact area similar to calf-raise. There is also some confusion between segmental rotation, bridge and chest press. In all of these exercises the user is lying on the back making the shape of the contact area similar. These exercises there are also subject to style differences between different users. Some users spread their arms more than others when performing bridge; some rotate more than others when performing segmental rotation; and some touch with the arms in the floor while doing chest presses while others move the arms in the air only.

The results presented so far are all are achieved by classifying exercises as individual, pre-segmented instances. However, usually exercises are grouped into sets where one set contains many of the same exercises without any segmentation. For further processing (counting as described in the next section) we need to be able to recognize which exercise such an unsegmented set belongs to. To this end we move a window over the complete exercises set. We choose a window size of 100 samples which is large enough to cover complete exercises. The window step is 10. For every step we classify the segment within the window. We then select the majority class as the class of the complete set. The evaluation was done as user independent study where the test subject was left out of the training data. Using this approach we reach 86.4% average accuracy for person independent classification of exercise sets. The correct class is occurring within the two majority classes 94.3% of the time.

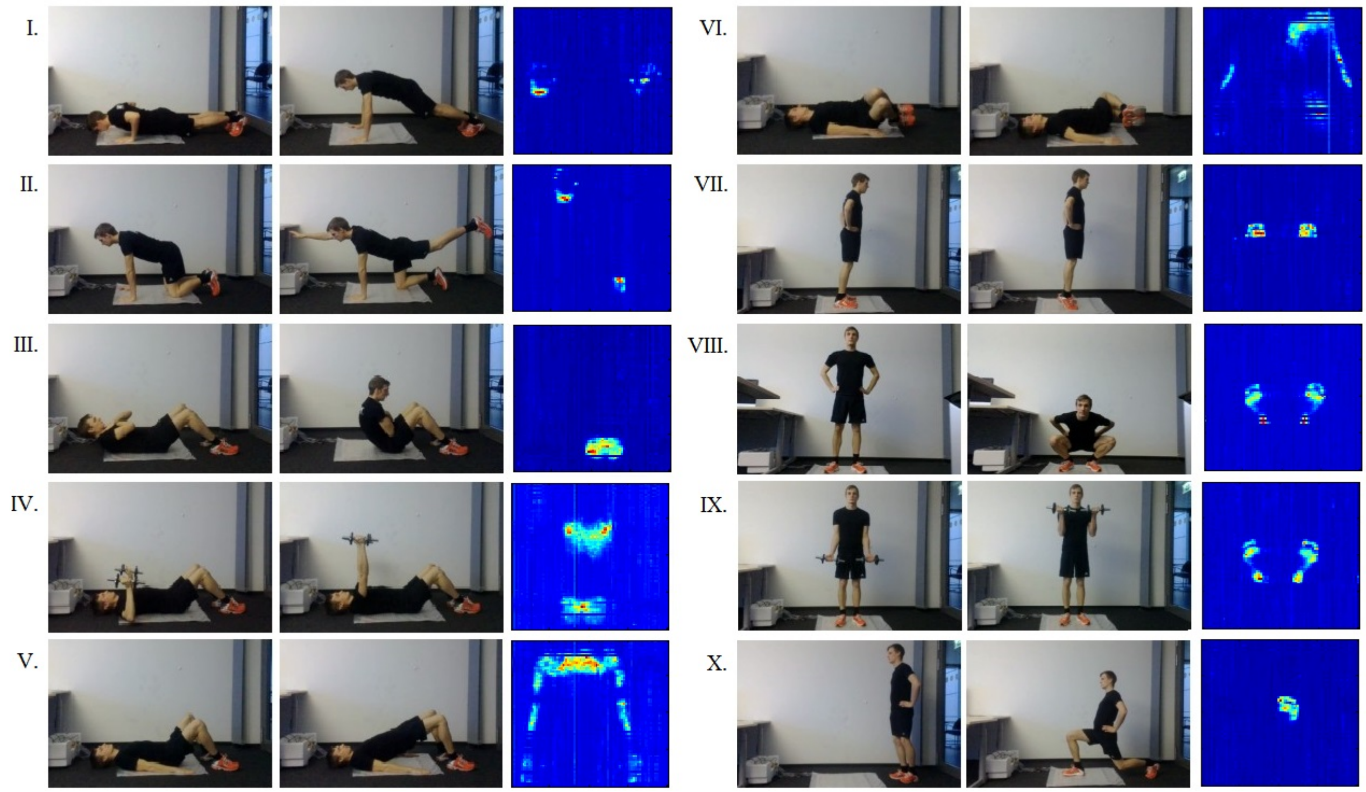


Figure 3: Exercises performed in the experiment: I. push-up, II. quadruped, III. abdominal crunch, IV. chest press with dumbbell, V. bridge, VI. segmental rotation, VII. calf raise, VIII. squat, IX. biceps curl with dumbbell, X. lunge. The pictures on the right are typical frames corresponding to the exercise.

In the next section we are investigating how exercises can be detected and counted within a set of exercises once the class to which they belong is known.

COUNTING REPETITIONS

In the previous section, it is shown that different exercises have distinguishable characteristics when looking at how the contact area and weight distribution varies over time. Ultimately, we would like to not only distinguish activities from each other, but also detect when they are performed, and to count how many the subject has performed within a set. This is achieved by calculating templates for each exercises and then match the data stream with the template. To make the system robust against variation in exercises speed we use dynamic time warping (DTW) when calculating the distance to the template. The following steps are performed:

1. Calculate area A, weight W, pressure p and Hu's Seven Moments ϕ_{1-7} for each frame in the recording.
2. Create exercise templates using training data.
3. Move the template over the test data and calculate a match at every step using dynamic time warping. A high peak in the match corresponds to good match of the template.
4. Detect and count peaks in the match to determine when and how many instances of an exercise have been performed in a set.

Template Calculation

As the initial step feature time-series $\mathbb{F} = \{A(t), W(t), p(t), \phi_{1-7}(t)\}$ are calculated for each frame in the recording. This step is the same as described in the previous section.

To describe how the template for an exercises is calculated, let $X_i(t) = \{\ X_1(t), X_2(t), X_3(t)\ ...\ X_k(t)\ \}$ be an arbitrary set of time-series representing instances of a specific exercise; k is the total number of instances of an exercise in the training data. The template length l is calculated as the median length of time series in $X_i(t)$. All exercises in $X_i(t)$ are then scaled by first upsampling and then downsampling so that they have the same length as the template. During upsampling intermediate pixels are calculated using linear interpolation. Before downsampling the signal is filtered to suppress aliasing. The initial template $T(t)$ is then calculated as the mean of all exercises in $X_i(t)$,

$$T(t) = \frac{\sum_{i=1}^{k} X_i(t)}{k}. \tag{1}$$

Labelled exercises might not only have different length but they might also be shifted in time. Therefore each exercise in $X_i(t)$ is shifted to yield the best alignment to the initial template $T(t)$. Cross-correlation between $T(t)$ and each exercise

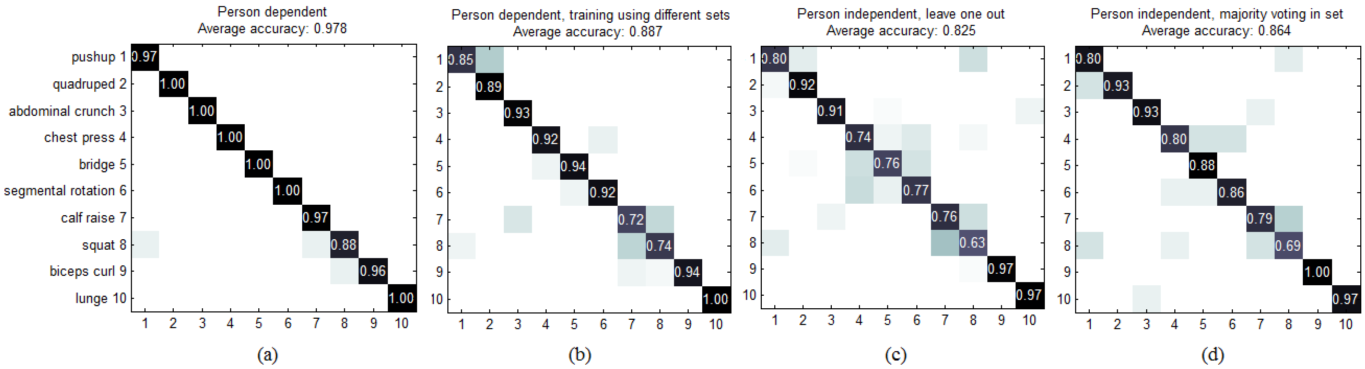


Figure 4: Classification results: (a) person dependent, (b) person dependent, trained and tested with different exercise sets, (c) person independent leave-one-out cross-validation, (d) person independent classification of a complete set of exercises using majority voting.

in $X_i(t)$ is used to determine how much an exercise should be shifted. The position where cross-correlation is the maximum is where $X_i(t)$ matches $T(t)$ the most. Shifted exercises are padded with the border values.

After the exercises have been correctly aligned the template is recalculated using Eq. 1. To further improve the template, each exercise in X_i is compared to the template, and a residual ε_i is calculated for each instance i as the sum of squared differences between each point in the time series and the template divided by σ^2,

$$\varepsilon_i = \sqrt{\sum_{t=1}^{l}(X_i(t) - T(t))^2/\sigma^2} \tag{2}$$

where σ is the standard deviation of the template. Weights w_i are calculated as the inverse of the residuals ε_i and normalized to yield a total sum of 1,

$$w_i = \frac{\frac{1}{\varepsilon_i}}{\sum_{i=1}^{k}\frac{1}{\varepsilon_i}}. \tag{3}$$

Exercises that are better aligned with the template will get a higher weight, and exercises that badly represent the template will get a lower weight. The template is recalculated as a weighted average of all exercises in $X_i(t)$,

$$T(t) = \frac{\sum_{i=1}^{k} w_i \times X_i(t)}{k}. \tag{4}$$

A template weight w is saved along with the template and calculated as

$$w = \frac{1}{\sum_{i=1}^{k}\varepsilon_i}. \tag{5}$$

Template weight is a value representing how good the template $T(t)$ is. When all exercises are lining up well, the residuals are small and the weight is large. A large weight means that the template is a good representation of the exercises. Templates and weights are calculated for each time series in $\mathbb{F}$. Different templates will have different weights depending on how well they represent the exercises. For example push-ups will cause large variation in pressure due to the up-down movement, while the area is the same during the whole exercise. Abdominal crunches on the other hand causes a large change in the area while the change in weight is not as dominant. The weights are normalized to 1. A complete template $\mathbb{T}$ of an exercise class is composed of:

1. 10 template time-series $T_s(t)$, $s \in [1, 10]$, one for each time series in $\mathbb{F}$.
2. 10 template weights w_s, $s \in [1, 10]$, one for each template time-series.

Dynamic Time Warping

To detect when a user performs an exercises we compare the incoming data stream with the template exercise. When calculating the distance between two time-series point-wise Euclidean distance is sensitive to both linear and non-linear temporal variation of the time-series. For example if two subjects perform exercises at different tempo matching data points will be temporally misaligned causing a large distance even though the exercises are the same. To overcome this problem we utilize dynamic time warping.

Dynamic time warping (DTW) is an algorithm that utilizes dynamic programming to find the optimum distance between time series [15]. DTW was first introduced by the speech recognition community, but it has later also proven to work well in other domains involving human motion [6, 9, 11]. In its simplest form DTW searches for the optimal distance between two time series A and B by constructing a distance matrix

$$D(i, j) = ||a_i - b_j|| \tag{6}$$

where a_i is *i*th element of time series A, $i \in [1, m]$, and b_j is the *j*th element of time-series B, $j \in [1, n]$. The optimal warping distance is that shortest accumulated distance path through the distance matrix $D(i, j)$ such that following conditions are met:

1. *Boundary condition:* the path starts at $D(1, 1)$ and ends at $D(m, n)$.
2. *Monotonic condition:* the path will not turn back, i and j indexes never decrease.
3. *Continuity condition:* the path advances one step at a time, i and j increases by maximum 1 at each step.
4. *Warping window condition:* the path must stay within a window size r from the diagonal.

Other constraints has been proposed to further restrict the warping path and to narrow down the search space, but for simplicity we are only using these four. The optimal path can be found using dynamic programming [15].

Matching Using Dynamic Time Warping

We perform the DTW matching in feature domain $\mathbb{F}$. A window with the same length as the template is moved over the time series of the test subject with 1 sample window step. At each step τ, the DTW distance between the time series segment within the window $Y(\tau)$ and the template $T(t)$ is calculated. Since $\mathbb{F}$ has 10 time-series the DTW is distance calculation is performed in 10 dimensions by using the Euclidean distance norm. Thus the DTW distance matrix becomes,

$$D(i, j) = \sqrt{\sum_{s=1}^{10} (T_s(i) - Y_s(j))^2}. \tag{7}$$

To emphasise the dimensions that contains more relevant information we incorporate the template weights w_s in the DTW distance calculations,

$$D(i, j) = \sqrt{\sum_{s=1}^{10} \frac{1}{w_s} \times (T_s(i) - Y_s(j))^2}. \tag{8}$$

Before the comparison, both the $T(t)$ and $Y(\tau)$ are normalized so that the mean is zero and standard deviation is one. The DTW distance depends on the both the template size and the length of the warping length of the distance matrix. To compensate the distance is normalized with the length of the warping path K. By calculating the DTW distance to the template at every window step we get the distance as a function of time step τ. The match $\mu(\tau)$ is inverse to the normalized DTW distance,

$$\mu(\tau) = \left[\frac{dtw(T(t), Y(\tau))}{K}\right]^{-1}. \tag{9}$$

Counting of Exercises

The time series $\mu(\tau)$ calculated in the previous section represents how well the test data matches the template at each time step τ. Peaks in $\mu(\tau)$ represent a good match to the template at that position. To count the exercises we define a static threshold. Every peak that is crossing the threshold is counted as an exercises. The peak detection algorithm can be described as following:

1. When the signal value is higher than threshold + tolerance, the signal enters a peak area.
2. When the signal value is lower than threshold - tolerance, the signal leaves the peak area.
3. The returned peak position is the highest peak inside the peak area.

The tolerance is used to avoid false peaks caused by a noisy signal. A low threshold will lead to higher detection rates but might also introduce false identifications, while a higher threshold will only detect exercises that are performed according to the template. In order to be able to compare the results we used the same threshold for every test subject and exercises.

To validate that the detection of an exercise is correct, the detected position is compared with the labelled start and end positions of an exercises. If the detected exercises is within the start and end position of the label it is considered as a match. To separate repeated exercises from each other we reduce the label size by 10%. Exercises that are detected in the border between exercises are not counted as match. Two detected exercises cannot be matched with the same label or vice versa. Detected exercises that match a label is considered *true positive*. Detected exercises that do not match any labels are considered *false positives*. Labelled exercises that are not detected are considered *false negatives*. Using this information the *precision* and *recall* can be calculated as,

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{10}$$

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{11}$$

The F_1*–score* is calculated as the harmonic mean between precision and recall,

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{12}$$

F_1–score is selected as the performance measure for our application since having few false positives and few false negatives are equally important.

Template Selection

As a final step we consider a situation where we have data from many users, each possibly doing the exercise in a

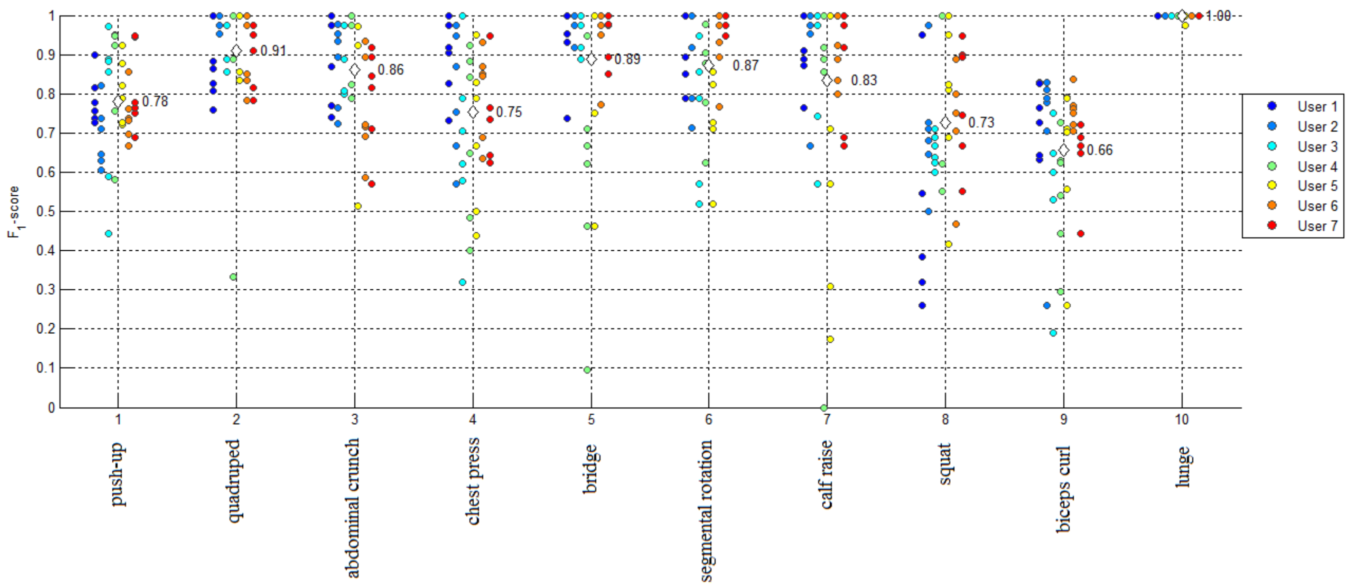


Figure 5: Detection rate of exercise counting for each subject and activity using different templates.

slightly different way. The question is how to best leverage the variability in the data to facilitate user independent recognition that is robust against strong variations in the way test users perform the exercises. We propose to construct different templates each well suited for a different user type. Given a new user to whom the system should be applied the system should automatically recognize which template is best suited and use it for counting. To this end we calculate the match (the same as used for counting) using all possible templates. We then select the template that has the highest match value during the whole set.

Results and Discussion

To evaluate the counting algorithm we calculate templates for each exercise and each subject. We then use the templates of one subject to detect exercises performed by the rest of the subjects. We repeated this using every subjects as the template. F_1-score is calculated for each exercise combination of subjects and templates. The results are presented in Figure 7.

The x-axis represents different exercises and the y-axis represents the detection score measured in F_1-score. Each dot is the resulting detection rate for each combination of subjects and templates. Different colors represent score for different users with different templates. The core for each user is slightly shifted horizontally for better visibility. The average for each exercises is marked as a white diamond. One can see that most dots are in the top quartile for all exercises. Lunges has an almost perfect detection score for all combination of subjects. This is explainable since the the variation in both speed and style between subjects is very low for this exercise. Biceps curl has lowest recognition score on average. This exercises is performed standing and pressure change is only caused by subtle change in the center of weight while the user moves his hands. The average F_1-score for all exercises is 82.8%. With respect to the practicability of user independent training we would like to note two points:

1. A few combinations of templates and subjects have a recognition score under 0.5. This illustrates the fact that there can indeed be strong differences in the way some exercises are performed by different user. Obviously, when the training and testing data come from such user pairs the performance is poor. However, such cases are rare with less than 5% of the points falling into this category.

2. The best case performance for each user is above 90% for most exercises. In other words, for every user there exists at least one other user whose exercise style is similar enough to facilitate good recognition. This suggests that inter user variability is a matter users belonging to a limited number of distinct "types" rather then each user doing the exercises in a different way. Thus, with a training set that contains all (most) types of user we can achieve good user independent performance and there is no need for user specific training.

Results with Template Selection

Achieving good results by having many different user types in the training set requires the ability to pick the best matching template for a given test user. The results of applying our template selection method are shown in Figure 6, which contains only scores produced using a training template from a user selected by our method. Comparison to Figure 5 shows that the average for all exercises improves significantly and is over or close to 0.9 in most cases. There are no results below 0.5, only 2 results below 0.6 and 3 more below 0.7. Of the 5 results below 0.7, 3 are from the same user who seems to have no well matching user type within the training set (which is quite small with just 7 test subjects). Overall the results are a strong indication that the general approach of using many training users and an automated template selection

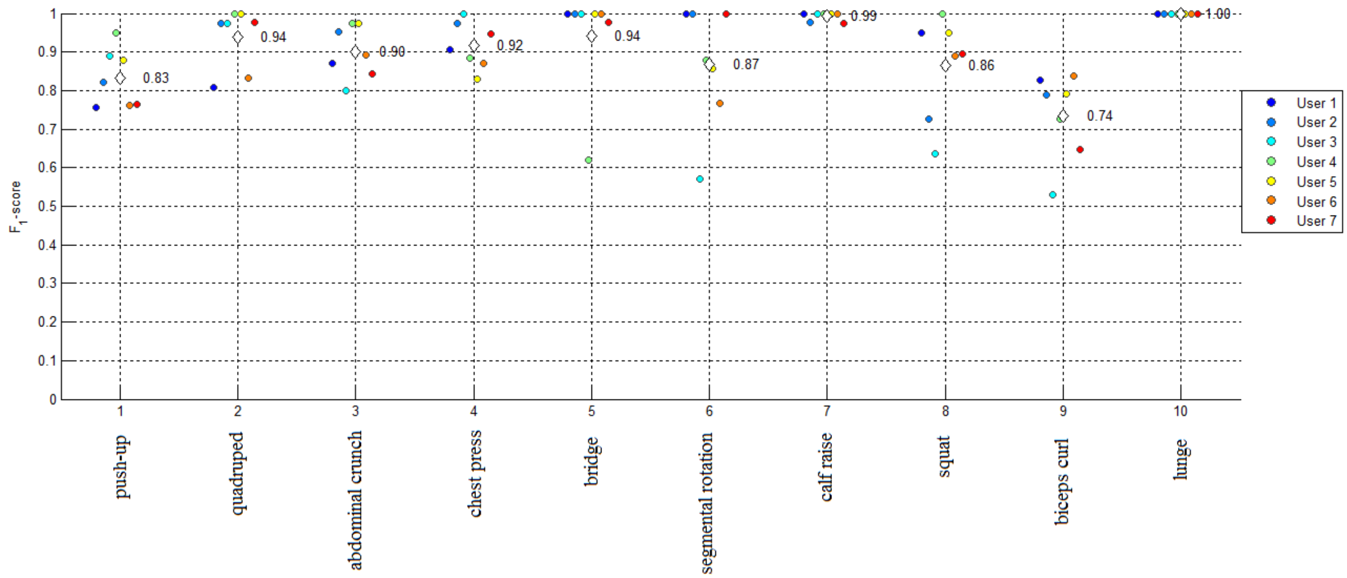


Figure 6: Detection rate of exercise counting for each subject and activity when using template selection.

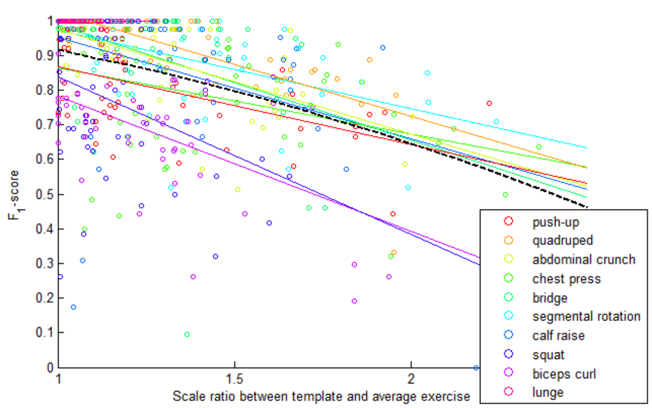


Figure 7: How ratio between exercise length and template length affects detection performance.

is a promising way for building a reliable user independent system.

Sensitivity to Speed Variations

To investigate if the influence of different exercise speeds on the accuracy of our system we plot in Figure 7 the detection score against the time ratio between template and the average exercise in the set. Different points indicate different combinations of users used for training and testing, and different colors indicate different exercise classes. We draw regression lines for each exercises class (colored lines), and a second order polynomial regression for all data points (black). We can see that when the ratio increases the performance is going down. However even with a scaling factor if 1.5–2 the detection score for most combinations of subjects and templates lies in the section of 0.7 and above. This is much less than the performance differences that can appear when the users belong to different "types" as discussed above. From the graph we can infer that when using a medium speed template (where the speed difference is less then 1.5 to any other template) scores are hardly affected by the speed. In order to cover an even larger range of different exercise speeds, multiple templates could be used.

CONCLUSION

The main conclusion of the work presented in this paper is that a textile pressure sensor matrix integrated in a gym-mat is a promising approach for monitoring strength related exercises that are performed on the ground. The accuracies that we have achieved (just under 90%) are well within the range typically seen in for example pedometers and could already be a useful practical tool. With respect to practical applications it is particularly important that we have achieved an 89.9% detection accuracy in a fully user independent way by leveraging a combination of DTW template matching and our template selection method.

ACKNOWLEDGEMENT

This work was partially supported by the collaborative project SimpleSkin under contract with the European Commission (#323849) in the FP7 FET Open framework. The support is gratefully acknowledged. The authors would also like thank all students and colleagues who volunteered as test subjects in our experiments.

REFERENCES

1. 2008 Physical Activity Guidelines for Americans. Tech. rep., U.S. Department of Health and Human Services, 2008. `http://www.health.gov/paguidelines/pdf/paguide.pdf`.
2. Fitbit: One & Zip Wireless Activity & Sleep Tracker, 2014. `http://www.fitbit.com/`.

3. FitLinxx: Professional Fitness Tracking System, 2014. `http://www.fitlinxx.net/`.
4. Mayo Clinic Fitness Multimedia, 2014. `http://www.mayoclinic.org/healthy-living/fitness/multimedia/HLV-20049447?s=9`.
5. Polar: Heart Rate Monitors and GPS Enabled Sport Watches, 2014. `http://www.polar.com/`.
6. Akl, A., and Valaee, S. Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing. *Acoustics Speech and Signal Processing (ICASSP)* (2010).
7. Annesi, J. J. Effects of computer feedback on adherence to exercise. *Perceptual and Motor Skills 87* (1998), 723–730.
8. Chang, K.-h., Chen, M. Y., and Canny, J. Tracking Free-Weight Exercises. In *Proceedings of the 9th International Conference on Ubiquitous Computing* (2007), 19–37.
9. Corradini, A. Dynamic time warping for off-line recognition of a small gesture vocabulary. In *In Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in RealTime Systems (RATFG-RTS01*, IEEE Computer Society (2001), 82.
10. Hu, M.-K. Visual pattern recognition by moment invariants, computer methods in image analysis. *IRE Transactions on Information Theory 8* (1962).
11. Kale, A., Cuntoor, N., Cuntoor, N., Yegnanarayana, B., Chellappa, R., and Rajagopalan, A. N. Gait analysis for human identification. In *LNCS* (2003), 706–714.
12. Khotanzad, A., and Hong, Y. H. Invariant image recognition by zernike moments. *IEEE Trans. Pattern Anal. Mach. Intell. 12*, 5 (May 1990), 489–497.
13. Muehlbauer, M., Bahle, G., and Lukowicz, P. What Can an Arm Holster Worn Smart Phone Do for Activity Recognition? *2011 15th Annual International Symposium on Wearable Computers*, 1 (June 2011), 79–82.
14. Pernek, I., Hummel, K. A., and Kokol, P. Exercise repetition detection for resistance training based on smartphones. *Personal and Ubiquitous Computing 17*, 4 (Dec. 2012), 771–782.
15. Sakoe, H., and Chiba, S. Readings in speech recognition. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990, ch. Dynamic Programming Algorithm Optimization for Spoken Word Recognition, 159–165.
16. Seeger, C., Buchmann, A., and Van Laerhoven, K. myHealthAssistant: A Phone-based Body Sensor Network that Captures the Wearer's Exercises throughout the Day. *Proceedings of the 6th International ICST Conference on Body Area Networks* (2011).
17. Zhou, B., Cheng, J., Sundholm, M., and Lukowicz, P. From Smart Clothing to Smart Table Cloth: Design and Implementation of a Large Scale, Textile Pressure Matrix Sensor. In *Springer Lecture Notes on Computer Science (LNCS) series*, ARCS 2014 - International Conference on Architecture of Computing Systems (2014).