

Detecting Unseen Anomalies in Weight Training Exercises

Yousef Kowsar¹, Masud Moshtaghi², Eduardo Velloso¹, Lars Kulik², Christopher Leckie²

¹Microsoft Research Centre for Social NUI
The University of Melbourne
y.kowsar@student.unimelb.edu.au,
evelloso@unimelb.edu.au

²Computing and Information Systems
The University of Melbourne
{masud.moshtaghi, lkulik,
caleckie}@unimelb.edu.au

ABSTRACT

In weight training, correct exercise execution is crucial for maximizing its effectiveness and for preventing injuries. However, given the complexity of these movements, it is a challenge for trainees to know whether they are performing the exercise correctly. Considering the fact that wrong moves may result in life long injuries, it is important to design systems that can detect incorrect performances automatically. In this paper, we present a workflow to detect performance anomalies from only observations of the correct performance of an exercise by the trainee. We evaluated our algorithm on a benchmark data set for the biceps curl exercise, and also evaluated our system with a publicly available dataset, and showed that our method detects unseen anomalies in weight lifting exercises with 98 percent accuracy.

Author Keywords

Weight training, activity assessment, user feedback, wearable computing, clustering, machine learning

ACM Classification Keywords

Human-centred computing → Ubiquitous and mobile computing → Ubiquitous and mobile computing systems and tools.

INTRODUCTION

Weight training is an effective way of building strength, increasing overall health, lowering the risk of diabetes and improving general fitness levels, among other benefits. Free weight exercises—such as those using dumbbells, barbells and kettlebells—can produce even better results when compared to machines. However, they require high *mechanical specificity*—appropriate movement patterns, force application, and velocity of movement (Stone et al., 2002)—in other words, a correct technique. An incorrect technique not only reduces the effectiveness of the exercise, but is also the number one cause of training injuries (Gallagher, 1996). This is even worse for free



Figure 1. Unilateral biceps curl performance. Top pictures show correct performance. Bottom row shows two variations. Left is a correct variation (with a green tick) and right is an incorrect variation (with a red cross).

weight exercises, which account for over 90% of weight-training injuries (Kerr et al., 2010).

The ubiquity of motion sensors makes them an appealing solution for offering automated feedback on users' exercise technique. However, current systems are limited to measuring heart rate and counting repetitions. Our goal is to build a system that can monitor a user's technique when performing free weight exercises to ensure correct technique and to prevent injuries.

A major challenge in automatically monitoring weight lifting exercises is the inherent complexity and high number of degrees-of-freedom of human movement. On one hand, the number of possible mistakes for any given exercise is huge. For example, in the exercise depicted in Figure 1, the trainee must lift the dumbbell all the way to the top, with the hips steady. Rocking the hips during the movement could cause serious back injury. On the other hand, any given exercise has multiple small variations to target specific muscle fibres that do not necessarily represent a mistake. For example, the exercise in Figure 1 could also be performed by twisting the dumbbell while lifting.

The inherent complexity of weight lifting exercises creates two specific challenges for any machine learning approach to recognise incorrect executions. First, each exercise is performed in sets of 6-15 repetitions, with each repetition being the unit of analysis. Therefore, accurately

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org. OzCHI '16, November 29 – December 2 2016, Launceston, TAS, Australia Copyright © 2016 ACM 978-1-4503-3673-4/15/12... \$15.00 <http://dx.doi.org/xx.xxxx/xxxxxxx.xxxxxxx>.

segmenting the repetitions is a crucial first step in the analysis pipeline. In this paper, we revisit the dataset recorded by Velloso et al. to demonstrate how an accurate segmentation can improve the mistake recognition performance (Velloso et al., 2013a). Second, the huge number of possible mistakes for each exercise makes a supervised learning approach for mistake recognition difficult (Velloso et al., 2013b). We propose an exercise analysis approach that infers a prototype of an exercise based only on its correct execution and distinguishes correct performance from incorrect ones with high precision and recall.

In summary, we propose a workflow for performance error detection in weightlifting exercises (see Figure 2). We therefore, contribute:

- 1) A mathematical model for repetition segmentation, along with its concrete implementation and evaluation, (Figure 2-A).
- 2) An algorithm that learns a prototype of an exercise from wearable sensor data, (Figure 2-B).
- 3) A statistical method for identifying incorrect executions based on deviations from the exercise prototype, (Figure 2-C).
- 4) An evaluation of our approach on a publicly available dataset that shows that our system is able to identify incorrect performance with 98% accuracy.

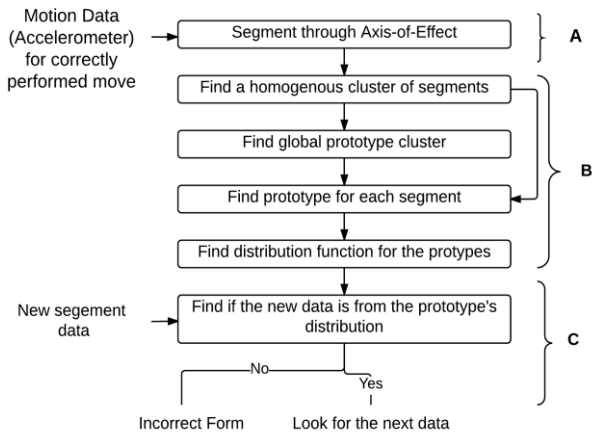


Figure 2. Workflow for finding incorrect moves in a weight lifting exercise.

RELATED WORK

There have been many studies to show how to perform a routine correctly to gain the best outcome from strength training. For a complete guide see (Brown 2007). Thus the knowledge of how to perform strength training routines correctly is available. Although trainees can read about how to perform an exercise and watch online resources, it is still very hard for them to know whether they are performing the exercise correctly or not.

In this regard, Chang et al. addressed the problem of detecting weight-lifting exercises using motion sensors (Chang et al. 2007). They showed that it is possible to use machine learning techniques to detect what the person has performed and count the number of repetition. Many studies have been applied the same techniques to detect

and count the number of repetitions. In 2013, NuActiv was designed to answer the problem of finding the unseen weight-lifting activities (Cheng et al. 2013). Although NuActiv was successful in detecting new routines it cannot measure the quality of the performance. RecoFit tried to detect multiple weight training exercises using a wearable sensor on the trainee’s wrist (Morris et al. 2014). A similar system named FEMO is designed to monitor free-weight exercise online using RFID technology (Ding et al. 2015). However, these systems are limited in telling whether the trainee is following the correct form of the exercise or not.

Detecting the errors in a weight training routine has been the focus of studies that assess the quality of any given performance. In one of the first attempts, Velloso et al. introduced the idea of classifying exercises by error type (Velloso et al. 2013a). They identified the most common posture errors during a specific exercise. Through an empirical study, they showed that using on-body motion sensors, it is possible to classify these error types performed by the user, using a classifier. However, they found that scaling this method to arbitrary mistakes is a challenge. Our method solves this shortcoming by introducing the idea of learning from correct performances. More recently Pernek et al. designed a hierarchical system for finding the intensity of the weight training exercises using machine learning tools (Pernek et al. 2015). Although their system can be extended to calculate the intensity of the performance it cannot differentiate between wrong performances and correct ones. MotionMA introduced the idea of learning from gym experts (Velloso et al. 2013b). The authors designed a system that can learn a move from experts such as personal trainers and monitors others during their performance to alert them, when some deviation detected. Their system is based on Microsoft Kinect. YouMove was designed to teach users how to perform an exercise using Microsoft Kinect by helping the user to mimic an experts move using a mirror like monitor (Anderson et al. 2013). However, the use of a projector/Kinect to monitor and give feedback to the user limits the applicability of the system in a real setting as it requires a pre-existing infrastructure.

In more recent research with a focus on knee/hip injury detection, Ahmadi et al. showed how classifying training session activities helps in both aspects of injury management and performance enhancement (Ahmadi et al. 2014). For a survey of activity recognition, see (Bulling et al. 2014; Lara and Labrador 2013).

With all the advances in detecting anomalies in weight-lifting exercises, the main challenge to design a system that can both learn the correct performance of a new move and inform users when deviating from the correct move is still unsolved. A successful system should give the users the freedom of performing the exercise whenever and wherever they want. In this paper we focus on building such a system through wearable sensors. Our method is different from the aforementioned articles in that: First, it is a personalised method that can be tuned for each user. Second, it learns the move through wearable sensors, where we do not have a view of the body as a whole.

UNDERSTANDING THE PROBLEM DOMAIN

The lack of a proper exercise technique can not only lead to poorer outcomes, but can also lead to serious injury. Adequate supervision from a trainer is an important strategy to monitor the techniques of trainees and make sure that they perform the exercises correctly. This trainer-trainee interaction usually follows a cyclical loop (Velloso et al., 2013a). First, the trainer designs a program for the trainee, according to their personal needs. The trainee performs the routine for a few weeks, while being monitored by the trainer. The trainer then identifies further strengths and weaknesses in their performance, which they use to design their next exercise routine.

However, not all gym-goers can afford a personal trainer at every session; instead, trainees often meet their trainer when it is time to redesign their program. Scenario 1 illustrates this problem:

Mark joins a gym for the first time. At the gym, he meets Jane, who was assigned as his personal trainer for his first session. Knowing that Mark's goals include increasing his overall strength, Jane designs a 6-week program including 6 different free-weight exercises, each to be performed in 3 sets of 10 repetitions. She demonstrates the correct execution of each exercise and after each demonstration she asks Mark to do a few repetitions to ensure that he understood it. She gives him feedback to improve his technique until reaching an acceptable performance level. After going over all the exercises, Mark feels confident that he understood them. However, next time Mark comes to the gym, without Jane's supervision, he is not quite sure whether his performance is correct.

This scenario illustrates how even though the trainee might have understood the correct technique, without adequate monitoring by a professional, his subsequent performance is prone to mistakes. This highlights an opportunity for wearable technology to fill this gap by ensuring that the trainee's technique is correct in the absence of the trainer.

Not only novice weight-lifters can benefit from wearable technology support, as Scenario 2 illustrates:

Ronnie is an experienced weight-lifter. He is constantly trying to push his limits at every gym session, progressively increasing the weights in his exercises. Though he demonstrates complete mastery of the technique using light weights, the heavier the weight, the more difficult it is to maintain a proper technique. The physical and cognitive overload of the heavy weights make it very difficult to keep his hips steady as he performs a deadlift and he ends up straining his back.

This scenario shows that depending on the weight being lifted, even an experienced lifter, who has previously shown good performance on a given exercise can make mistakes in subsequent ones. The feedback given by a wearable system has the potential to help them perform exercises with heavy weights with a similar technique as when they lift light weights.

Combined, these scenarios suggest three important design challenges and opportunities. First, when a trainer teaches an exercise, though she may point out common mistakes, it is infeasible to record every possible mistake. Therefore, a system should be able to analyse further performances of an exercise based only on a correct execution. Second, given the multiple possible variations for a given exercise depending on the specific needs of individual trainees, a system should be able to learn a user-specific model of the exercise, tailored to each particular execution style. Third, we can assume that at a certain point the trainee is able to perform the exercise with the correct technique (under the supervision of the trainer in Scenario 1 and with a light weight in Scenario 2). Therefore, we are able to train a classifier with the data from a given user to evaluate subsequent performance of the same user.

In the following sections, we present our approach to exercise analysis based on these principles. We first revisit the dataset of Velloso et al. (2013b) and demonstrate how an improved segmentation procedure can improve mistake detection in a supervised learning approach. We then propose a novel analysis method based only on the correct execution of an exercise that does not require the demonstration of mistakes, but is still able to detect them.

SEGMENTING REPETITIONS

Segmentation is the process of finding each individual repeat in a given time-series. Since our focus is on each individual repeat we need to find a way to correctly extracting each repeat from a given accelerometer time-series data.

Velloso et al. studied the possibility of classifying form correctness in weight training through unilateral biceps curls (Velloso et al. 2013a). They took a supervised learning approach, by recording both the correct execution of the exercise, as well as common types of mistakes. Because we use their dataset to evaluate our approach, in this section, we briefly describe the dataset and propose a new segmentation approach that improves the recognition accuracy using the same classifier used by those authors.

Dataset

The dataset of Velloso et al. includes the data from 6 participants performing 10 repetitions of the unilateral biceps curl exercise with 5 variations (Velloso et al., 2013a). The dataset is publicly available in the UCI Machine Learning Repository¹.

A unilateral dumbbell curl is a weight training exercise focused on strengthening the biceps muscle. The description of the exercise is as follows and is illustrated in Figure 1 (top row):

- Stand with a dumbbell in each hand.
- Keep the upper arm stationary, while bringing one of the forearms up until it reaches its maximum bend.
- Return the arm to its original position slowly.
- Repeat the same move with the other arm.

¹ [UCI Machine learning dataset: Wight Lifting Exercise Monitored](https://archive.ics.uci.edu/dataset/201/weight+lifting)

The authors defined four common errors happening during this exercise as follows: Class B: Moving upper arm to the front. Class C: Lifting the dumbbell half way up and return. Class D: Lowering the dumbbell halfway down. Class E: Pulling the forearm with the help of the hips at the start to lift the dumbbell. The correct form of the exercise is labelled Class A, accordingly. For each class label, they asked 6 participants to perform the exercise according to its class label specification. They collected the data using 4 sets of sensors placed on the glove, upper arm, dumbbell and belt. Each sensor set contains one three-axis accelerometer, one three-axis gyroscope and one compass. Each participant performs 10 repeats of the exercise for each label. For the detailed explanation of the dataset collection method see Velloso et al. (2013a).

Pre-processing

Any motion sensor will show some degree of white noise which comes from the nature of the sensor. White noise shows itself as small perturbations around the actual value. Any successful analysis of the data must first remove this noise from the data (Yun et al. 2006). In the literature the smoothing effect of a Kalman Filter has been praised as a high pass filter that can remove the noise with a high accuracy (Kalman 1960; Jun et al. 2006). In this project, we used a density based Kalman Filter implementation (Byron et al. 2004). Since we are trying to remove white noise, we configured the Kalman filter with low sensitivity to the current read and high sensitivity to previous reads by setting the deviation of the current reads to be 10 times the deviation of the previous reads. The result is shown in Figure 3.

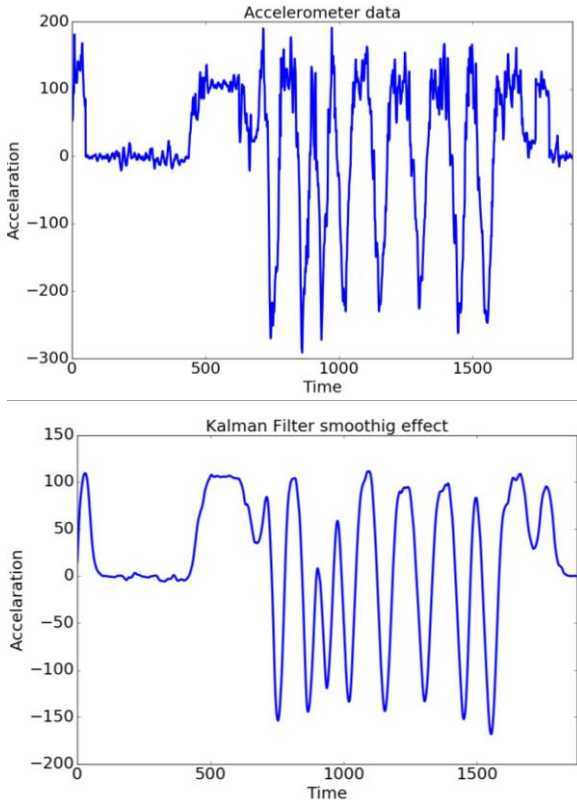


Figure 3. Smoothing accelerometer with Kalman Filter.
Top figure shows the raw data from accelerometer.
Bottom figure shows the result of Kalman Filter on the same data.

We draw the readers' attention to the peak parts from the raw data. The accelerometer shows a very sudden high acceleration and drop, which is due to the effect of stopping the dumbbell. It is clear how the Kalman Filter reduces this effect and smooths the result.

Since the data is published separately, for consistency we went through all the labelling to verify that the labels are correct. We found issues with the correct execution of 2 of the participants. By definition, label-A data should have a steady pattern from the belt accelerometer data. However, in two cases, the participants incorrectly moved their hips during the exercise, as illustrated in Figure 4. We therefore discarded the data from these two participants from our subsequent analyses.

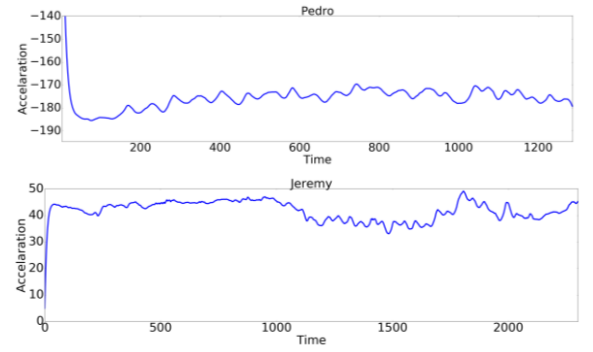


Figure 4. High perturbation area for Jeremy makes his class label "A" invalid. For a correct label for class "A" we expect a steady series like the one in the top figure. Note that the actual repeats start at around *time* = 800 for Jeremy.

Segmentation

Weight training exercises are mostly repetitive tasks where trainees perform the same move for a few repetitions, i.e., the move starts from a starting point, follows a path in space and returns to the starting point again. Each exercise set will contain 6-15 of these repetitions. Therefore, the collected data from motion sensors will show a cyclical pattern in the time series. An example for a unilateral biceps curl acceleration data can be seen in Figure 5.

As a result, correctly identifying the move using the motion-graph requires finding its starting point in the acceleration graph. To formulate a move, define " f " to be the time-movement function showing the path for the move in space. Thus f is a function of change-of-position in x, y and z direction. See equation 1.

$$f = g(x, y, z) \quad (1)$$

As a result, we need to define function g to describe the move. Since finding the actual move is highly dependent on all three dimensions and very sensitive to any noise/error in the data, we followed the method described in (Mortazavi et al. 2014). They showed that in weight training exercises motion can be mainly captured from a single axis in space. We call this axis the axis-of-effect (AoE). For example, "y-axis" is the AoE in the unilateral biceps curl exercise (Figure 6).

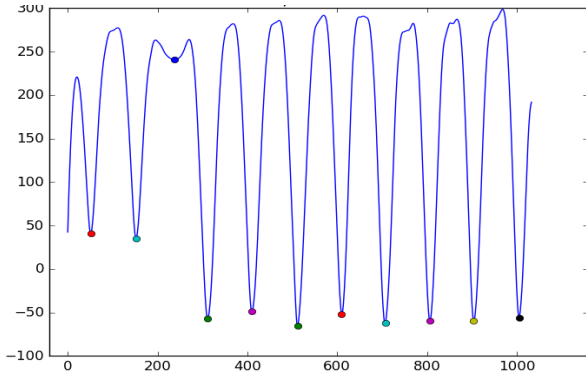


Figure 5. Motion sensors will show a repetitive pattern for weight lifting exercises. In this figure, the repetitive pattern is clear from the accelerometer attached to the trainee's forearm, while performing a unilateral biceps curl.



Figure 6. y-axis is the Axis-of-Effect (AoE) in unilateral biceps curl. The actual motion happens in y direction (drawn in red).

The unilateral biceps curl in the AoE direction starts with a positive acceleration to lift the weight up. Then it follows a negative acceleration, which results in stopping the dumbbell at the peak. It is then followed by a negative acceleration pattern to bring down the dumbbell, which follows a positive acceleration to bring the dumbbell back to the steady point. Therefore, function f can be estimated by $g(a_{AoE}(t))$ where $a_{AoE}(t)$ is the acceleration function in the AoE direction and t is time.

$$f \approx g(a_{AoE}(t))$$

Considering the unilateral biceps move, we are searching for points in the function f where the function has reached its optimum values. That is:

$$\frac{df}{dt} = 0$$

Using the chain rule:

$$\frac{df}{dt} = \frac{dg}{da} \frac{da}{dt} = 0$$

Since changes in acceleration can describe the changes in movement, we can estimate the optimum points by only considering $\frac{da}{dt} = 0$. That is:

$$\frac{df}{dt} = 0 \approx \frac{da}{dt} = 0$$

We look for points in the acceleration-time graph where the derivatives went to zero, or equivalently we are looking for points where the derivatives of the acceleration-time

function changes sign. Since we are looking only for start points of each move, we only look for minimum points on the acceleration-time graph, which lets us segment the repetitions (indicated by dots in Figure 5).

To evaluate the accuracy of our model, we manually annotated the acceleration-time graph and compared the manually annotated segments with the automatically detected segments. Table 1 shows the performance of our model for finding the repeats in acceleration-time graph.

	Precision	Recall
Average	0.965	0.82

Table 1. The average precision and recall for the segmentation algorithm.

The result shows that our algorithm finds the correct segmentation points with high precision. The recall value shows that the algorithm finds more segments than necessary. Filtering the points by the average repetition size improves the recall to 90%, which is an acceptable rate.

Classification

In Velloso et al. (2013b), the authors reported the best classification accuracy achieved by segmenting the data using a fixed sliding window of length 2.5 seconds. This result comes from the fact that each repeat will take around 2 seconds. Therefore, setting the window size to 2.5 seconds will capture the entire move. However, a fixed window size will result in two main problems. First, a fixed window size captures overlapping repetitions, resulting in missing the starting and ending part of a repeat. Missing the start or end of a repeat will stop any system from correctly alerting users as soon as they start deviating from a correct form. Second, a fixed window size is highly dependent on the person and the exercise. For example, if the user performs the exercise too quickly, a fixed window might capture two or more repetitions. Given that our unit of analysis is each individual repetition, it is crucial to capture the whole repetition with no extra data.

For comparison, we applied the same classifier proposed by Velloso et al., only using our segmentation method. The result is presented in Table 2. The result shows that our proposed algorithm not only provides a method that can easily be generalized, but also boosts the classification task.

	A	B	C	D	E
Optimum point	57.3	56.3	56.7	56.4	56.8
Fixed window	52.1	54	53.5	53.7	53.2

Table 2. The average precision of the classification task in percentage. The top row is from data segmented by our method. The bottom row is from a fixed window size.

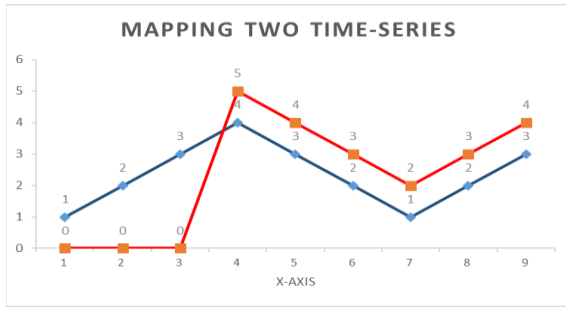


Figure 7. Mapping two time-series.

Discussion

These results show that in a supervised learning scenario, where we have data for the correct execution of an exercise, as well as data for each possible repetition, segmenting the data using windows that precisely match the repetition improves the classification performance. However, it is unlikely that in a realistic use case we would have data for every possible mistake. Given that an incorrect form might result in lifelong injury, the stakes for the problem are high enough that we need a system that can robustly detect previously unseen mistakes only based on the correct form of the exercise. In such systems, a model should be created from the correct moves. After learning the correct move, every move is compared with the ground truth model to detect deviations from the model. Designing such a model is not a straightforward task. There are many parameters involved in a ground truth model such as the height of the person, the weight used for the exercise, the duration of the repeat, etc. In the next section, we show how we can derive a personalized model from the correct moves in the exercise we are studying, a ground truth model for the unilateral biceps curl.

DERIVING A GROUND TRUTH MODEL

The two scenarios we presented above suggest that at some point the trainee will be able to demonstrate a correct performance, either because he is under the supervision of a trainer or he is using a lighter weight. Using this assumption, we can derive a user-dependent ground truth model specific to the trainee's needs that they can use in subsequent repetitions (when the trainer is absent or with a heavier weight) to evaluate their performance.

Data collection

In this phase the personal trainer makes sure the person is capable of correctly performing the exercise and initiates data collection. We use the class label A, from the dataset of Velloso et al. as our correct data.

Pre-processing

As discussed earlier, the recorded data for an exercise routine includes multiple repeats of same exercise. However, each routine starts and ends with recordings, which are usually related to releasing or carrying the weight, not related to the actual routine. To clear Class A segmentation, we used a clustering algorithm to put the segments with high similarity into the same group. This task is very important because we can make sure the ground truth method is only generated from homogenous segments and prune from any anomalies itself. We

continued the clustering algorithm until we had a cluster of size “number of repeats - 2”. This value is selected because we are aware that starting and ending segments might have extra movements that make them different from the segments for the rest of the repeats. We used the single linkage clustering algorithm to cluster our segments. For the distance in the single linkage clustering we used two metrics described in the next section.

Distance metrics

Distance from two time-series can be defined by mapping every point of one time-series to another point in the other time-series. Then the distance is the result of summation over every two pairs in the two time-series. See Figure 7 and equation 2. Figure 7 shows one of the possible maps for the two time-series. The points with the same x-value are mapped to each other. By defining the distance function, $dist$, as the distance between two pairs the distance between two time-series $X = (x_1, \dots, x_n)$ and $Y = (y_1, \dots, y_n)$ is defined as:

$$p = \sum_{i=1}^n dist(x_i, y_i) \quad (2)$$

We are looking for a map that minimizes p . The distance matrix for a set of time series will be a matrix where each cell represents the minimum distance between the time series indexed by the row and the column of the matrix.

There are many possible options for a distance function. The k-shape algorithm suggests that the normalized cross correlation metric is a reliable choice. We are using the same function for our experiment.

Ground Truth Prototype

Deriving a prototype for time-series data is a challenging task. The main issue is how we can map one time-series to another. The main method for such a mapping is based on finding the minimum distance from mapping each point from one time-series to another. Dynamic Time Warping (DTW) (Muller 2007) has been traditionally used to perform this task. More recently, the K-shape (Paparrizos and Gravano 2015) algorithm has been introduced with promising results. Both approaches will derive a comparison based method to define a distance between time-series data. By clustering the time-series data, Paparrizos and Gravano showed how to design an algorithm that can find a trajectory which satisfies the minimum sum distance between its points and all the other points in the associated time-series cluster.

Paparrizos and Gravano showed that by knowing a distance matrix, we can reduce the problem of finding the prototype for a cluster to a maximum likelihood problem where the eigenvectors of the Hessian matrix will define the prototype for the cluster. They named their algorithm shape-extraction, which we will adopt in this paper. For a detailed argument see the original paper shape (Paparrizos and Gravano 2015).

In this paper, we use the same method for deriving a prototype trajectory from the correctly performed moves. We then use a statistical method to capture any deviation

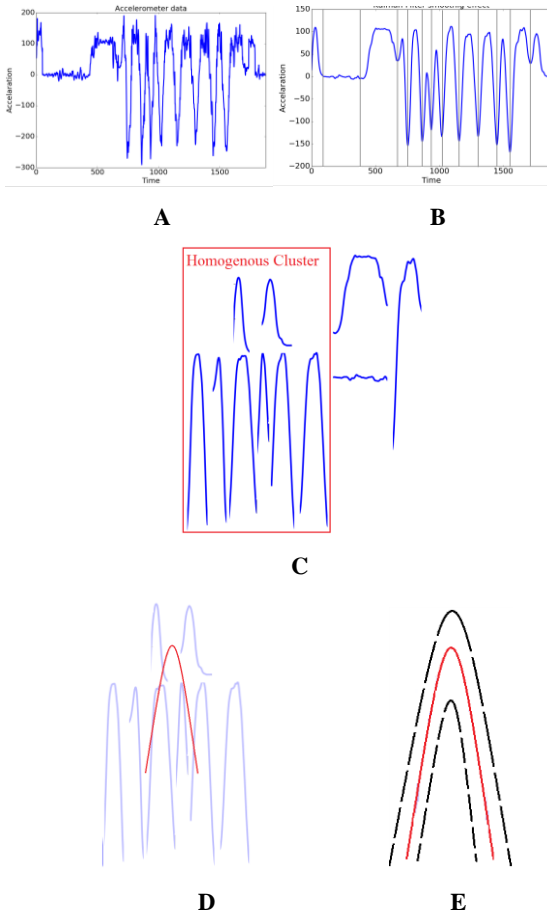


Figure 8. Finding weightlifting anomalies workflow in a nutshell. A. Collecting raw data for correct performance. B. Smoothing the raw data. C. Creating a homogenous cluster of correct segments. D. Deriving the ground truth (Red line). E. Calculating standard deviation for the ground truth (Dashed line)

from this prototype. First we briefly describe k-shape and DTW metrics and how we can derive the trajectory.

To find the ground truth trajectory, we developed Algorithm 1. For each person in the dataset, we passed all the homogenous segments clustered in pre-processing part from the class A dataset to the shape-extraction algorithm and find the ground truth trajectory (Personal GTT).

Finding anomalies

Using Personal GTT, we first enumerate each segment from each person and calculate the associated trajectory for that segment. Then, for each point in time in the trajectory set we find the mean and standard deviation among all calculated trajectories. Receiving any new segment for the person we find the trajectory for the new segment according to the related Personal GTT. We then compare each point in the new trajectory to be with in the $mean \pm 3s.d.$ range. Three standard deviations is selected from the 3 sigma rule which states that nearly all values from a distribution occur within the range of three standard deviations from the mean value (Grafarend 2006). If any point is found outside this margin from the new trajectory, we label it as a wrong form segment. See Algorithm 2.

Input: N: New segment; G: Global prototype from Algorithm 1; M: mean for G (from algorithm 1); S: standard deviation for G (from algorithm 1)

Output: True if the new segment is an anomaly otherwise False

```

1: function FIND_ANOMALY(N, G, M, S)
2:   shape ← extract_shape([N, G])
3:   for each p ∈ shape do
4:     ▷ shapep is shape's value at index p
5:     ▷ Sp is S's value at index p
6:     ▷ Mp is M's value at index p
7:     if shapep > Mp + 3 * Sp or shapep < Mp - 3 * Sp
8:       then
9:         return True
10:    end if
11:  end for
12:  return False
end function

```

Algorithm 1. Finding the prototype for the master class. Personal class segments come from clustering similar segments of the correct class in the pre-processing phase.

Since anomalies may be found from different sensors—for example when the trainee is moving their hip the belt accelerometer will detect the mistake—we applied the same algorithm (find-anomalies) for each time-series from the available accelerometer sensors attached to belt, arm and forearm. We define a deviation from the correct form as any deviation from any time-series from any sensor. This way our system can also detect why the person is deviating from a correct form. For example, in the test study for the unilateral biceps curl, the system tells whether the trainee is moving their hip or their arm when they were not supposed to. The system also detects when during the segment the user deviated from the correct form, i.e., whether it was in the first quarter of the move, in the middle or in the last quarter. These two sets of information not only help users maintain the correct form but also let personal trainers to design more personalised routines that consider the strength and weakness of each trainee. For a brief overview of the output of each step in the workflow see Figure 8. The workflow starts by collecting some data for the correct performance of the move (sample data is

Input: C: cluster of homogenous segments for correct move performed by the person

Output: Prototype for given cluster, the mean and deviation of the calculated prototype according to the member of the cluster.

```

1: function GROUND_TRUTH(C)
2:   globalTrajectory ← extract_shape(C)
3:   trajectory ← []
4:   mean ← []
5:   sd ← []
6:   for each s ∈ C do
7:     shape ← extract_shape([s, globalTrajectory])
8:     trajectory.append(shape)
9:   end for
10:  for each p ∈ globalTrajectory do
11:    ▷ trajectoryp is the pth value of all trajectories
12:    m ← mean[trajectoryp]
13:    s ← standard_deviation(trajectoryp)
14:    mean.append(m)
15:    sd.append(s)
16:  end for
17:  return globalTrajectory, mean, std
18: end function

```

Algorithm 2. Checking anomalies for new data

shown in Figure 8-A). We remove all the white noise from the collected data using a Kalman filter and segment the data using our proposed segmentation algorithm (Figure 8-B). We create a homogenous cluster from the segments created in part B. (Figure 8-C). We derive the prototype for the ground truth performance using the homogenous cluster (red line in Figure 8-D). We calculate the standard deviation for our prototype using the correct performance (dashed line in Figure 8-E). Note that the homogeneity of segments in part C will make sure that only the correct performances of the exercise are considered for deriving the ground truth in part D and not moves with an extra part such as the very first move in which the data contains the part where the trainee is picking the weight at the very beginning of the routine.

Results

To test our approach, we used the same dataset and two metrics, namely dynamic time warping and shape-based. We considered class label A as the correctly performed class and used the rest of the classes as the test cases. For testing our ground truth trajectory, we manually labelled each correct segment in class A and feed them to Algorithm 2. The result can be seen in Table 1.

Both algorithms can reliably detect mistakes for the Unilateral Biceps Curl. However, it is clear that shape based distance can perform the task better in finding the trajectory. This is mainly because of the way k-shape algorithm finds distances, which highlights the correlation among the points in two time-series. Both algorithms have shown some false positives which are mainly for the segments at the start or end of the routine. This is mainly because the start and end of a routine is very hard to correctly segment. It is often the case that the segmentation has considered an extra part to the beginning of the segment for the starting segment or considered an extra part for the ending segment. These mis-segmentations result in False Positives in our algorithm.

DISCUSSION

In this paper we designed a system that can learn an activity from the user and monitor the person to perform the weight lifting task. Our main focus for this research was on weight training exercises. However, the scope of our system is not limited to weight lifting activities. We would like to see the performance of our system in other applications where users must perform some repetitive task and their performance needs to be monitored, such as physiotherapy, physical rehabilitation, elderly fitness, health insurance, etc. In physiotherapy and in post-surgery in general, doctors will ask their patients to perform some daily routines and monitor their progress. It is often the case that the patients will learn a routine at the hospital and then should follow the same routine at home for a few weeks or months. Our system can be seen as provide the foundation for doctors to monitor their patients' progress.

In recent years we have seen scenarios where insurance companies try to promote the health of their clients by encouraging them to be more active. In these scenarios insurance companies can use our system to help their clients perform strength training routines correctly which

	Shape Base Distance		Dynamic Time Warping	
	TP	FP	TP	FP
Pedro	41/42	0/10	40/42	1/10
Carlitos	44/44	0/10	43/44	0/10
Charles	44/44	2/10	44/44	3/10
Eurico	40/40	2/10	40/40	3/10

Table 1. True Positive (TP) and False Positive (FP) for anomaly detection algorithm. The cells with bold font show the winner algorithm for detecting anomalies.

has been shown to improve the health and wellbeing of their clients.

CONCLUSION

In this paper, we designed a system that detects incorrect moves from learning only the correctly performed routine. We showed why correctly segmenting each repetition during a weight lifting exercise is important. We designed a mathematical model that can detect the start and end point of each repetition, while the user is performing the exercise. We designed our workflow based on the segments from correctly performed routines. This lets us use our system in an online environment where the system can detect any anomalies as soon as the end of a repetition is detected.

Our workflow starts by correctly segmenting the time-series data using the data from an Axis-of-Effect accelerometer. It calculates the prototype from the segments using the extract-shape method proposed by Paparrizos and Gravano (Paparrizos and Gravano 2015). Using the derived prototype, the system finds the distribution of the trajectory from mapping each segment to the prototype. Finally, for each new segment it checks whether the new data's trajectory to the ground truth is from the calculated distribution or not. If not, the algorithm rejects the segment and alerts the trainee of an anomaly.

Since the only input argument for calculating the prototype is correctly segmented temporal data, our method can be easily generalized to any motion sensors, observing other parts of the body during the exercise. We just need to segment the relative data according to the Axis-of-Effect sensor. Afterwards, the workflow can find anomalies for any sensor attached to the trainee's body. This generalization lets us detect any anomalies from movement in any other parts of the body, which in turn helps both trainees and trainers to work closely together to achieve their goals.

In this paper we designed a system that both beginners and professional weight lifters can benefit from. Beginners can learn how to perform a correct routine under a personal trainer's supervision. Our system will help them maintain the correct form while exercising on their own. Professionals can train our system by performing the correct routine in a safe environment, like using light weights. Then our system will help them maintain the correct form while exercising under more difficult conditions, such as at the end of the session or with heavier weights.

REFERENCES

- Ahmadi, A., Mitchell, E., Destelle, F., Gowing, M., O'Connor, N. E., Richter, C. & Moran, K. (2014), Automatic activity classification and movement assessment during a sports training session using wearable inertial sensors, In Proc. '14 11th International Conference on Wearable and Implantable Body Sensor Networks', ACM Press (2014)-98–103.
- Anderson, F., Grossman, T., Matejka, J. & Fitzmaurice, G. (2013), YouMove: Enhancing movement training with an augmented reality mirror, In Proc. 26th Annual ACM Symposium on User Interface Software and Technology', UIST '13, ACM Press (2013)- 311–320.
- Brown, L. E. (2007), 'Strength training: National strength and conditioning association', Champaign, IL: Human Kinetics.
- Bulling, A., Blanke, U. & Schiele, B. (2014), 'A tutorial on human activity recognition using body-worn inertial sensors', Journal of ACM Computing Surveys. 46(3), 33:1–33:33.
- Byron, M. Y., Shenoy, K. V. & Sahani, M. (2004), 'Derivation of extended Kalman filtering and smoothing equations', URL: http://www-npl.stanford.edu/b~yronyu/papers/derive_eks.pdf 19, 25.
- Chang, K.-H., Chen, M. Y. & Canny, J. (2007), Tracking free-weight exercises, In Proc. 9th International Conference on Ubiquitous Computing', Springer-Verlag (2007) -19–37.
- Cheng, H., Sun, F., Griss, M. L., Davis, P., Li, J. & You, D. (2013), Nuactiv: recognizing unseen new activities using semantic attribute-based learning, In Proc. MobiSys, ACM Press (2013)- 361–374.
- Ding, H., Shangguan, L., Yang, Z., Han, J., Zhou, Z., Yang, P., Xi, W. & Zhao, J. (2015), FEMO: A platform for free-weight exercise monitoring with RFIDs, In Proc. 13th ACM Conference on Embedded Networked Sensor Systems', ACM Press (2015)-141–154.
- Gallagher, M. (1996), 'Ten most common causes of training injury', Muscle & Fitness.
- Grafarend, E. W. (2006), Linear and nonlinear models: fixed effects, random effects, and mixed models, Walter de Gruyter.
- Jun, J., Guensler, R. & Ogle, J. (2006), 'Smoothing methods to minimize impact of global positioning system random error on travel distance, speed, and acceleration profile estimates', Transportation Research Record: Journal of the Transportation Research Board (1972), 141–150.
- Kalman, R. E. A new approach to linear filtering and prediction problems. Journal of basic Engineering, (1960), 82(1), pp.35-45.
- Kerr, Z. Y., Collins, C. L. & Comstock, R. D. (2010), 'Epidemiology of weight training-related injuries presenting to united states emergency departments, 1990 to 2007', The American Journal of Sports Medicine 38(4), 765–771.
- Lara, O. D. & Labrador, M. A. (2013), 'A survey on human activity recognition using wearable sensors', IEEE Communications Surveys Tutorials 15(3), 1192–1209.
- Morris, D., Saponas, T. S., Guillory, A. & Kelner, I. (2014), Recofit: Using a wearable sensor to find, recognize, and count repetitive exercises, In Proc. SIGCHI Conference on Human Factors in Computing Systems', ACM Press (2014) - 3225–3234.
- Mortazavi, B. J., Pourhomayoun, M., Alsheikh, G., Alshurafa, N., Lee, S. I. & Sarrafzadeh, M. (2014), Determining the single best axis for exercise repetition recognition and counting on smartwatches, In Proc. International Conference on Wearable and Implantable Body Sensor Networks', IEEE (2014)-33-38.
- Paparrizos, J. & Gravano, L. (2015), k-shape: Efficient and accurate clustering of time series, In Proc. SIGMOD International Conference on Management of Data', ACM Press (2015),-1855–1870.
- Pernek, I., Kurillo, G., Stiglic, G. & Bajcsy, R. (2015), 'Recognizing the intensity of strength training exercises with wearable sensors', J. of Biomedical Informatics 58(C), 145–155.
- Stone, M., Plisk, S. & Collins, D. (2002), 'Training principles: evaluation of modes and methods of resistance training—a coaching perspective', Sports biomechanics/International Society of Biomechanics in Sports 1(1), 79–103.
- Velloso, E., Bulling, A. & Gellersen, H. (2013a), Motionma: Motion modelling and analysis by demonstration, In Proc. SIGCHI Conference on Human Factors in Computing Systems', ACM Press (2013)-1309–1318.
- Velloso, E., Bulling, A., Gellersen, H., Ugulino, W. Fuks, H. (2013b), Qualitative activity recognition of weight lifting exercises, In Proc. 4th Augmented Human International Conference', ACM Press (2013),-116–123.
- Yun, X. & Bachmann, E. R. (2006), 'Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking', IEEE Transactions on Robotics 22(6), 1216–1227
- Müller, M., 2007. Dynamic time warping. Information retrieval for music and motion, pp.69-84.
- Paparrizos, J. and Gravano, L., May. k-Shape: Efficient and accurate clustering of time series. In Proc. ACM SIGMOD International Conference on Management of Data, ACM Press (2015),-69-76
- Pernek, I., Kurillo, G., Stiglic, G. and Bajcsy, R., 2015. Recognizing the intensity of strength training exercises with wearable sensors. Journal of biomedical informatics, 58, pp.145-155.
- Stone, M., Plisk, S., and Collins, D. Training Principles: Evaluation of Modes and Methods of Resistance Training – A Coaching Perspective. Sports biomechanics/International Society of Biomechanics in Sports 1.1 (2002): 79-103.

- Velloso, E., Bulling, A., Gellersen, H., Ugulino, W. & Fuks, H. Qualitative activity recognition of weight lifting exercises. In Proc. 4th Augmented Human International Conference (2013a), ACM Press (2013)-116-123.
- Velloso, E., Bulling, A. & Gellersen, H. MotionMA: motion modelling and analysis by demonstration. In Proc. SIGCHI Conference on Human Factors in Computing Systems (2013b). ACM Press (2013)-1309-1318.
- Yun, X. and Bachmann, E.R. Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking. IEEE Transactions on Robotics (2006), 22(6) - 1216-1227.