AMIE: Automatic Monitoring of Indoor Exercises

Tom Decroos[1(✉)], Kurt Schütte[2], Tim Op De Beéck[1], Benedicte Vanwanseele[2], and Jesse Davis[1]

[1] Department of Computer Science, KU Leuven, Celestijnenlaan 200A, Leuven, Belgium
{tom.decroos,tim.opdebeeck,jesse.davis}@cs.kuleuven.be
[2] Department of Movement Sciences, KU Leuven, Tervuursevest 101, Leuven, Belgium
{kurt.schutte,benedicte.vanwanseele}@kuleuven.be

Abstract. Patients with sports-related injuries need to learn to perform rehabilitative exercises with correct movement patterns. Unfortunately, the feedback a physiotherapist can provide is limited by the number of physical therapy appointments. We study the feasibility of a system that automatically provides feedback on correct movement patterns to patients using a Microsoft Kinect camera and Machine Learning techniques. We discuss several challenges related to the Kinect's proprietary software, the Kinect data's heterogeneity, and the Kinect data's temporal component. We introduce AMIE, a machine learning pipeline that detects the exercise being performed, the exercise's correctness, and if applicable, the mistake that was made. To evaluate AMIE, ten participants were instructed to perform three types of typical rehabilitation exercises (squats, forward lunges and side lunges) demonstrating both correct movement patterns and frequent types of mistakes, while being recorded with a Kinect. AMIE detects the type of exercise almost perfectly with 99% accuracy and the type of mistake with 73% accuracy. Code related to this paper is available at: https://dtai.cs.kuleuven.be/software/amie.

1 Introduction

Being active is crucial to a healthy lifestyle. Initiatives such as *Start to Run* [3] in Belgium and *Let's Move* in the USA [28] encourage people to become more active. These initiatives are paying off, as in the USA, almost every generation is becoming more active, according to a report made by the Physical Activity Council [9]. However, this increase in activity inevitably also leads to an increase in sports-related injuries [12,24]. Besides the short and long term physical discomforts, there are substantial costs associated with injuries. A significant portion of these costs are allocated to rehabilitation [13,18]. People with injuries usually need to visit a physiotherapist. The physiotherapist will then prescribe a program of rehabilitation exercises that the injured patient must follow at home.

U. Brefeld et al. (Eds.): ECML PKDD 2018, LNAI 11053, pp. 424–439, 2019.
https://doi.org/10.1007/978-3-030-10997-4_26

This current rehabilitation paradigm has several drawbacks. First, due to time constraints of the patients, and the cost of physiotherapy sessions, the interaction between the patient and physiotherapist is necessarily limited. Second, many patients simply do not do their exercises [4], with research reporting adherence rates to home exercise programs of only 15–40% [5,15]. Third, it is hard for a patient to learn how to correctly perform the exercise due to the limited feedback by a physical therapist. These drawbacks can cause problems such as prolonged recovery time, medical complications, and increased costs of care [22].

One possible way to address these drawbacks is to exploit technological advances to develop an automated system to monitor exercises performed at home. Patients have expressed a willingness to use such a system because it allows them to perform exercises in the comfort of their own home while having fast access to feedback [19]. Such a home monitoring system could provide three important benefits, by:

1. Motivating the patient to do his exercises;
2. Showing the patient the correct movement patterns for his exercises; and
3. Monitoring the quality of the performed exercises and giving feedback in case an exercise is poorly executed.

Currently, most effort has been devoted towards addressing the first two tasks. First, researchers have shown that home-systems can successfully motivate patients to adhere to their home exercise programs by applying tools such as gamification and social media [14,17,27]. Second, several approaches have demonstrated the ability to show the correct movement patterns of exercises in a clear way such that people are able to understand and reproduce these movement patterns [7,19,25]. There has been less work on the third task: monitoring the correctness of exercises. The current approaches typically make unrealistic assumptions such as the availability of perfect tracking data [26], fail to describe how the system determines if an exercise is performed correctly [7,15], or do not quantitatively evaluate their systems [1,10,26,29,30].

In this paper, we propose AMIE (Automatic Monitoring of Indoor Exercises), a machine learning pipeline that uses the Microsoft Kinect 3D camera to monitor and assess the correctness of physiotherapy exercise performed by a patient independently in his home. At a high-level, AMIE works as follows. First, it identifies an individual repetition of an exercise from the Kinect's data that tracks the absolute location of multiple joints over time. Second, in order to capture the movement of the patient, AMIE rerepresents the time-series data for an exercise with a set of simple statistical features about the angles between interconnected joints. Finally, it detects an exercise's type, correctness, and, if applicable, mistake type. We evaluated AMIE on a data set of 1790 exercise repetitions comprising ten different test subjects performing three commonly used rehabilitation exercises (i.e., squat, forward lunge and side lunge). AMIE detects what exercise is being performed with 99.0% accuracy. In terms of predicting the correctness of an exercise and which mistake was made, AMIE achieves accuracies of 73.4% and 73.8% respectively.

To summarize, this paper makes the following contributions:

1. Details the data collected in this study comprehensively;
2. Discusses a number of challenges related to representing the Kinect data;
3. Describes the entire pipeline for classifying exercises correctly, including how to automatically detect an individual exercise repetition and how to predict if an exercise is performed correctly;
4. Assesses AMIE's ability to (a) detect the exercise being performed, (b) determine if the exercise was performed correctly, and (c) identify the type of mistake that was made; and
5. Releases both the collected data set and the source code of AMIE at http://dtai.cs.kuleuven.be/software/amie, as a resource to the research community.

2 Data Collection

We describe the characteristics of the subjects who participated in this study and the collected data.

2.1 Subjects

Data of 7 male and 3 female subjects (26.7 ± 3.95 years, 1.76 ± 0.12 m, 73.5 ± 13.3 kg, 23.38 ± 2.61 BMI) were collected. All subjects were free of injuries and cardiovascular, pulmonary and neurological conditions that impeded the ability to perform daily activities or physical therapy exercises. The study was conducted according to the requirements of the Declaration of Helsinki and was approved by the KU Leuven ethics committee (file number: s59354).

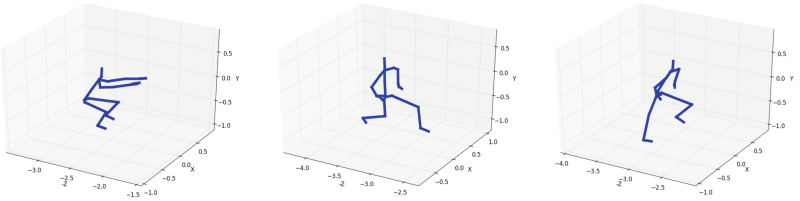
2.2 Exercises

The subjects were instructed to perform three types of exercises, which are illustrated in Fig. 1:

Squat. The subject stands with his feet slightly wider than hip-width apart, back straight, shoulders down, toes pointed slightly out. Keeping his back straight, the subject lowers his body down and back as if the subject is sitting down into a chair, until his thighs are parallel to the ground (or as close as parallel as possible). Next, the subjects rises back up slowly.

Forward lunge. The subject stands with his feet shoulder's width apart, his back long and straight, his shoulders back and his gaze forward. Next, the subject steps forward with his left (or right) leg into a wide stance (about one leg's distance between feet) while maintaining back alignment. The subject lowers his hips until his forward knee is bent at approximately a 90° angle. Keeping his weight on his heels, the subject pushes back up to his starting position.

Side lunge. The subject stands with his feet shoulder's width apart, his back long and straight, his shoulders back and his gaze forward. Next, the subject steps sideways with his right leg into a wide stance while maintaining back alignment. Keeping his weight on his heels, the subject pushes back up to his starting position.



(a) (b) (c)

Fig. 1. Figures recorded by the Kinect of a person doing (a) a squat, (b) a forward lunge and (c) a side lunge.

2.3 Mistake Types

In addition to correct repetitions of each exercise, the subjects were instructed to perform repetitions that illustrate common incorrect ways to perform each exercise. Specifically, we consider the following types of mistakes:

Squat Knees Over Toes (KOT). The subject executes a squat, but while lowering his back, the subject goes beyond alignment so that the knees go far over the toes.

Squat Knock Knees (KK). The subject executes a squat, but while lowering his back, the subject collapses his knees inward.

Squat Forward Trunk Lean (FTL). The subject executes a squat, but while lowering his back, the subject tilts his trunk forward, so that his back is no longer straight or perpendicular to the ground.

Forward lunge KOT. The subject executes a forward lunge, but while stepping forward, the subject goes beyond alignment so that the knees go far over the toes and the forward knee is bent further than a 90° angle.

Forward lunge KK. The subject executes a forward lunge, but while stepping forward, the subject collapses his forward knee inward.

Forward lunge FTL. The subject executes a forward lunge, but while stepping forward, the subject tilts his trunk forward, so that his back is no longer straight or perpendicular to the ground.

Side lunge FTL. The subject executes a side lunge, but while stepping sideways with his right leg, the subject tilts his trunk forward, so that his back is no longer straight or perpendicular to the ground.

For side lunges, only the forward trunk lean was performed because the other two mistakes are not applicable to this exercise.

2.4 Protocol

The study protocol was designed in collaboration with biomechanics researchers with extensive expertise in collecting and analyzing data of rehabilitation exercises. Each subject was instructed to perform six sets of ten repetitions of each type of exercise (squat, forward lunge, and side lunge). Given sufficient rest in between the sets, the subjects were asked to perform ten repetitions of the exercise within a set one after the other, while briefly returning to an anatomical neutral position between repetitions. Each set was monitored using a Kinect (Microsoft Kinect V2) that is able to capture the movement of 25 joints at 30 Hz. The Kinect was positioned such that the subject was facing the Kinect at a distance of 1–2 m.

More specifically, the subjects were asked per exercise to first perform three sets of ten correct repetitions. Before the first set of each exercise, the correct execution was explained and demonstrated by a physiotherapist. In case of the forward lunge, the subjects were asked to alternate between stepping forward with their left and right leg between executions. Next, the physiotherapist demonstrated mistakes that are often made by patients while executing these three exercises. For squats and forward lunges, the *KOT*, *KK*, and *FTL* mistakes were demonstrated and the subjects performed one set of ten repetitions of each. For side lunges, only the *FTL* mistake was demonstrated as the other two mistakes are not applicable. The subjects performed three sets of ten repetitions of the *FTL* mistake in case of the side lunges, because we wanted to collect the same number of recorded repetitions per exercise.

3 The AMIE System

Our goal is to develop a Kinect-based system that provides automatic feedback to patients. Such a system requires performing the following steps:

1. Extracting the raw data from the Kinect;
2. Partitioning the stream of data into individual examples;
3. Rerepresenting the data into a format that is suitable for machine learning;
4. Learning a model to predict if an exercise was done correctly or not; and
5. Providing feedback to the user about his/her exercise execution using the learned model.

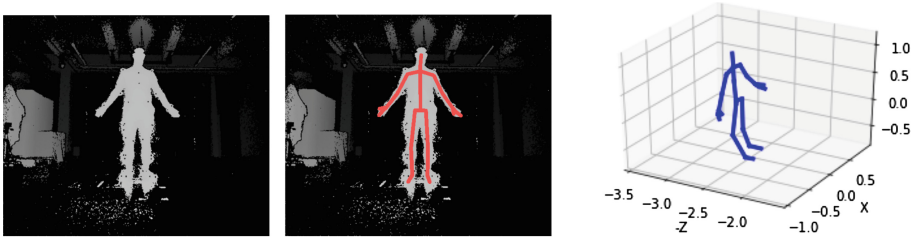
In this paper, we study whether detecting if an exercise was performed correctly or not is feasible. We establish a proof of concept called AMIE (Automatic Monitoring of Indoor Exercises) that currently addresses only the first four tasks.

3.1 Extracting the Kinect Data

The Kinect records a set of exercise repetitions as a video, which consists of a sequence of depth frames. A depth frame has a resolution of 512×424 pixels where each pixel represents the distance (in millimeters) of the closest object

seen by that pixel. Using the Kinect's built-in algorithms, each depth frame can be processed into a stick figure (Fig. 2).

Each set of repetitions is stored as a XEF file (eXtended Event Files), which is a format native to the Kinect SDK that can only be interpreted by applications developed in the closed software system of the Kinect. It cannot be directly examined by conventional data analysis tools such as Excel, R, and Python. Through manual examination of the Kinect SDK and some reverse engineering, we have developed a tool that takes as input a Kinect video of n depth frames stored as a XEF file and outputs a sequence of n stick figures in JSON format. This tool is freely available at http://dtai.cs.kuleuven.be/software/amie.



(a) (b) (c)

Fig. 2. (a) A depth frame as shown in KinectStudio, an application for recording and viewing Kinect videos built on the Kinect SDK (b) the stick figure built by the Kinect's algorithms as shown in KinectStudio (c) the same stick figure extracted from the native Kinect file format and plotted using Python, a popular data analysis tool.

All ten subjects performed roughly six sets of ten repetitions for three exercises. Our data set D contains exactly 186 videos $v \in D$. We sometimes have more than 18 videos (3 exercises $\times$ 6 execution sets) per subject, because a video recording of ten repetitions could get broken up in two separate videos due to technical issues.[1] Hence, not every video contains exactly ten executions.

Each video $v \in D$ can be represented as a tuple

$$v = ([f_i]_{i=0}^{n}, s, e, m)$$

where $[f_i]_{i=0}^{n}$ is a sequence of n stick figures $f_0, \ldots, f_n$, s is the identifier of the subject performing the exercise, e is the exercise type (squat, forward lunge or side lunge), and m is the mistake type. The mistake type m is *KOT*, *KK*, *FTL* or *None*. *None* means the exercise was performed correctly. A stick figure f_i is a vector of 25 joints. Each joint is represented by (x, y, z) coordinates, where z represents the distance to the Kinect camera and x and y represent respectively

[1] The subject were, in addition to the Kinect, also tracked with a Vicon camera system using reflective markers attached to the body. Due to sweat and movement, these markers sometimes fell off and the exercise repetition set was interrupted to reattach a marker. The collected Vicon data is not used in this paper.

horizontal and vertical positions in space. Examples of joints are the left ankle, the right knee, the left elbow, the spine base and middle, the left shoulder, etc.

3.2 Partitioning a Set of Exercises into a Single Repetition

Each video v in our data set D contains a sequence of stick figures $[f_i]_{i=0}^{n}$ that represents multiple repetitions of an exercise. This is problematic because we need to work on the level of an individual repetition in order to ascertain if it was performed correctly or not. Therefore, a sequence containing one set of k executions needs to be subdivided into k subsequences, with one subsequence for each repetition. We employed the following semi-supervised approach to accomplish this:

1. We select a reference stick figure f_{ref} that captures the pose of being in-between executions, such as in Fig. 3a. Typically, such a pose can be found as either the start or end position in the sequence.
2. We convert the original sequence of stick figures $[f_i]_{i=0}^{n}$ into an equal length 1-dimensional signal $[d(f_{ref}, f_i)]_{i=0}^{n}$, where the i^{th} value of the new signal is the distance d between the reference stick figure and the i^{th} stick figure in the original sequence. The distance d between two stick figures is the sum of the Euclidean distances between the 25 joints of each stick figure.
3. This new signal has a sine-like shape (Fig. 3b), because the distance between a stick figure in-between repetitions and f_{ref} is small whereas the distance between a stick figure in mid-exercise and f_{ref} is high. The valleys in the signal (i.e., the negative peaks) represent likely points in time when the subject is in between repetitions. To detect the valleys, we employ a modified version of the peak-finding algorithm in the signal processing toolbox of Matlab.[2] These valleys are used to subdivide the original sequence series into a number of subsequences, where one subsequence encompasses one repetition.
4. Depending on the quality of the resulting subsequences in the previous step, we do some additional manual modifications, such as inserting an extra splitting point or removing some stick figures from the start or the end of the original sequence.

Using our approach, we transformed the dataset D of 186 videos into a new dataset D' that contains 1790 repetitions $r \in D'$. The last step in our semi-supervised approach was necessary only for 15 out of 186 videos.[3]

Each repetition r is represented by a 4-tuple $([f_i]_{i=0}^{n}, s, e, m)$ just like a video v. The difference is that the sequence of stick figures $[f_i]_{i=0}^{n}$ of each repetition $r \in D'$ now only contains one repetition of the performed exercise instead of multiple repetitions. The length of the stick figure sequence per repetition ranges from 40 to 308 (136 ± 37).

[2] Our peak-finding algorithm takes the minimal peak distance as input, which we estimate from the data using the length of the sequence and the dominating frequency in the Fourier transform.

[3] Manual modifications were typically needed if the video recording was cut off too late, adding extra noise at the end.

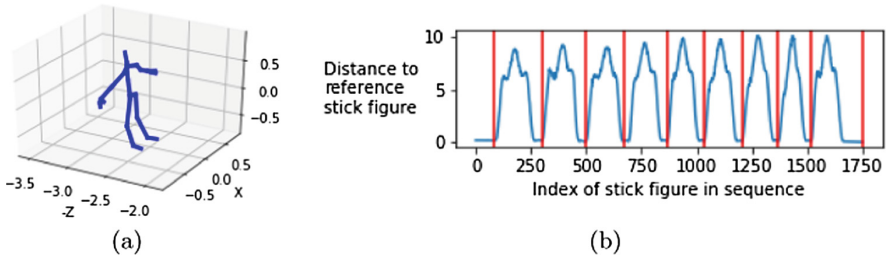


Fig. 3. (a) The reference stick figure. (b) The blue signal shows the distance between the reference stick figure and each stick figure in a sequence containing ten repetitions of an exercise. The automatically generated split points are highlighted with red vertical lines. (Color figure online)

3.3 Feature Construction

Two main challenges prevent us from applying an off-the-shelf machine learning algorithm directly to our data, its heterogeneity (e.g., examples are recorded from different subjects) and temporal nature (e.g., examples are sequences of varying length). We detail two consecutive transformations that address these challenges and construct a feature vector for each repetition r. We refer to these transformations as the heterogeneity transformation and the temporal transformation.

Heterogeneity Transformation: Not only do the subjects differ in height and weight, but also their relative position and orientation to the Kinect camera vary from exercise to exercise. All these variations affect the absolute coordinates recorded by the Kinect, but are independent to the executed exercise and its correctness. Therefore, we aim to remove these variations from the data, by using the geometrical angles in the interconnected joint triplets instead of the absolute coordinates. For example, *(left hip, left knee, left ankle)* is an interconnected joint triplet because the knee connects the hip and the ankle. Its geometrical angle is the angle formed at the *left knee* joint enclosed by the *(left hip, left knee)* and *(left knee, left ankle)* segments. For each stick figure f_i, the angles of all 30 such interconnected joint triplets are used as features. An additional advantage is that our new representation mimics the language physiotherapists often use to describe whether an exercise is performed correct or wrong (e.g., a good forward lunge has the forward knee bent at 90°).

Temporal Transformation: This transformation maps the representation of an exercise repetition from the variable-length sequence of highly self-correlated stick figures (represented by angles) to a fixed length feature vector. We observe that the temporal relationship between stick figures is not important to our tasks, because the exercises and mistakes can be recognized by specific poses. For example, the exercises are recognizable by the poses in Fig. 1, and a *KOT* mistake is made if a stick figure's knees go far over its toes. Moreover, only a subset of the stick figures need to be specific to an exercise or mistake type

to label the entire exercise repetition. Following this insight, our tasks can be framed as Multiple Instance Learning (MIL) problems. In MIL, learners have access to bags of examples, and each bag containing multiple examples. A bag is labeled positive if one or more of the contained examples is positive and negative otherwise [23]. In our case, a bag contains all the stick figures of one exercise repetition. We employ a popular approach for dealing with multiple instance learning: a metadata-based algorithm where the metadata for each bag is some set of statistics over the instances in the bag. This way the bags are mapped to single-instance feature vectors and the classification task can then be performed by an arbitrary single-instance machine learning algorithm [11,23]. Our set of statistics aims to describe the value distribution for each interconnected joint triplet angle using the following five summary statistics: minimum, maximum, mean, median and standard deviation. Each exercise is therefore mapped to a fixed length feature vector of 150 summary statistics (30 angles $\times$ 5 statistics).

3.4 Model Learning

AMIE learns three separate models, one for each task we consider:

T1: Identifying which exercise the patient is performing.
T2: Predicting whether the exercise was performed correctly or not.
T3: Detecting the type of mistake that was made when performing the exercise.

The first task may not strictly be necessary as a home monitoring system could ask the patient to perform the exercises in a set order. However, the ability to detect automatically which exercise is being performed would give the patient more autonomy when conducting his rehabilitation and would allow him to dynamically decide on and update his exercise routine.

Given that we have represented our data in a fixed-length feature format, it is possible to solve each of these learning problems using standard, off-the-shelf machine learning techniques. We tested five popular algorithms and found XGBoost [8] to be the most suitable. We provide further details on the process in the following section.

4 Experiments

The goal of the empirical evaluation is to address the following six research questions:

Q1: Can we accurately detect what exercise is being performed?
Q2: Can we accurately detect whether the exercise was performed correctly?
Q3: Can we accurately detect what type of mistake was made?
Q4: How does our classification approach compare to using hand-crafted rules and a nearest-neighbor approach?
Q5: Can our pipeline provide feedback to a patient in real-time?
Q6: Is the accuracy of AMIE dependent on the type of mistake?

4.1 Evaluation Methodology

When evaluating how well our learned models will generalize to unseen data, special care has to be taken in our setting to account for two types of dependencies that appear in our data. The first dependency arises due to the temporal nature of the data. An individual example is a single repetition of an exercise, but that repetition is done in a set of ten consecutive repetitions. Hence it will be correlated to the other examples in that set. The second dependency arises because one subject performs multiple repetition sets. Consequently, standard cross-validation is unsuitable as a repetition from the same set (or subject) may appear in both the train set and the test set, which could lead to over-optimistic accuracy estimates.[4] Therefore, we see two possibilities for performing cross-validation.

Leave-one-set-out cross-validation. In this setting, the data of one repetition set appears in the test set and the data of all other 185 repetition sets appears in the training set. Practically, this setting estimates the accuracy of a system that would only be deployed to monitor patients if it had examples of them performing the specific exercises that they must complete at home.

Leave-one-subject-out cross-validation. In this setting, the data for nine subjects appears in the training set and the data for the remaining subject appears in the test set. Practically, this setting estimates the accuracy of a system that can be deployed without collecting any data about the new patient performing his exercises. In other words, a system that is trained based on a fixed set of subjects and then deployed on new (i.e., previously unseen) subjects.

For each of our research questions, we consider both setups.

4.2 Results for Q1 Through Q3

Research questions Q1 through Q3 correspond to evaluating our accuracy on tasks T1 through T3. The learners we considered are Logistic Regression, Naive Bayes, Decision Tree, Random Forest, and XGBoost [8]. For all learners, we performed no hyperparameter tuning to avoid overfitting due to our limited amount of data. That is, we used the standard parameter settings from the Scikit-learn [20] and XGBoost [8] Python packages.

We trained models for each learner on all three classification tasks using both cross-validation schemes (Table 1). On our data, we can almost perfectly determine which exercise is being performed with all models. However, further investigation is needed to determine if this result holds when confronted with a wider range of exercise types, particularly for exercises that exhibit highly similar movement patterns.[5] When determining the correctness and mistake of

[4] Preliminary research suggests that the standard cross-validation setting indeed leads to an over-optimistic accuracy estimate.

[5] For example, a normal squat and a single-leg-squat exhibit similar movement patterns which could confuse our learner.

an exercise, XGBoost performs best under both cross-validation settings with an accuracy of at least 73%.[6] While we perform significantly better in T3 than random guessing (25%) or predicting the majority class (50%), we deem AMIE's current accuracy insufficient to be used as an autonomous system without the supervision of a physiotherapist.

Table 1. Accuracy of AMIE and baselines for different tasks, learners and cross-validation settings. We can almost perfectly identify the type of exercise (T1) with every learner. XGBoost generally performs the best at detecting correctness (T2) and mistake type (T3).

Task		T1		T2		T3	
Cross-validation setting		Set	Subject	Set	Subject	Set	Subject
AMIE	Decision Tree	0.992	0.973	0.731	0.671	0.642	0.555
	Logistic Regression	0.999	0.989	0.772	0.708	0.726	0.672
	Naive Bayes	0.982	0.972	0.633	0.646	0.478	0.547
	Random Forest	0.997	0.987	0.762	0.700	0.705	0.675
	XGBoost	0.997	**0.990**	0.790	**0.734**	**0.741**	**0.738**
Baselines	NN-DTW (absolute coord.)	**1.000**	0.965	**0.840**	0.623	0.627	0.555
	NN-DTW (angles)	0.997	**0.990**	0.713	0.648	0.576	0.549
	Handcrafted Rule Set	X	X	0.634	0.634	0.590	0.590

4.3 Results for Q4

We compared AMIE against two popular approaches in the literature: a nearest-neighbor approach using Dynamic Time Warping as a distance measure (NN-DTW), and a rule set handcrafted by a biomechanics researcher.

NN-DTW: This baseline is based on the work of Su et al. [25], who provide feedback on rehabilitation exercises using the distance of the executed exercise to a library of reference exercises. We employ the NN-DTW baseline using two different representations of our stick figures: the initial representation with the absolute (x, y, z)-coordinates of 25 joints and the representation using the geometrical angles in the interconnected joint triplets, which is obtained after applying the heterogeneity transformation as detailed in Sect. 3.3.

Handcrafted Rule Set: This baseline is inspired by Zhao et al. [29,30], who introduce a system that allows physiotherapists to express 'correctness rules'. Our rule set consists of three rules, one for each mistake. For example, the *KOT* rule states that if both the left and right knee joints have a z-coordinate that is closer to the camera than the z-coordinates of the left and right toes, then the subject is performing a *KOT* mistake. The *KK* and *FTL* mistakes are encoded in a similar way. To reduce the effect of noise in the data, we only predict an

[6] All learners employ only one model to detect mistakes for all three exercises. We tried learning one model per exercise, but noticed no difference in accuracy.

exercise repetition to have a specific mistake if at least ten stick figures in the repetition show that mistake. If this is not the case, the repetition is predicted to be correct. If multiple mistakes are detected in the exercise repetition, then the mistake with the most supporting stick figures is predicted.

The results of our baselines are shown in the lower half of Table 1. Except for one occurrence, AMIE (using the XGBoost classifier) always outperforms both the NN-DTW baselines and the handcrafted rule set on T2 and T3. This suggests that to provide accurate feedback, we cannot rely purely on domain experts, as a more flexible approach than a handcrafted rule set is necessary. However, we also cannot blindly apply machine learning techniques; NN-DTW is the most popular way to classify time series [2], yet it performs significantly worse than AMIE.

4.4 Results for Q5

The machine learning pipeline AMIE consists of four steps: (1) extracting the raw data from the Kinect, (2) partitioning the stream of data into individual examples, (3) constructing a feature vector for each example, and (4) detecting the examples' correctness using the trained models. Extracting the raw data from the Kinect into the JSON format and loading it in Python takes 0.15 s for one repetition set. Partitioning one repetition set into individual examples takes 0.03 s on average. Constructing the feature vectors takes 0.05 s on average per repetition set and detecting correctness (i.e., predicting the labels for *T1* through *T3* using our trained models) takes 0.0001 s on average per repetition set. In summary, a patient receives feedback from AMIE within 0.28 s after performing his exercises, which is almost instantaneous for human subjects. In addition, the largest fraction of our processing time is due to unnecessary disk I/O that could be avoided in a commercial implementation of AMIE. In this way, a patient can immediately adapt his or her incorrect movement patterns to the correct movement patterns, therefore accelerating the learning process of rehabilitation and mimicking a real-life scenario where a focused physiotherapist typically provides expert feedback after a few repetitions in practice.

4.5 Results for Q6

To check whether the accuracy of AMIE is dependent on the type of mistake, we inspect the confusion matrices for detecting mistake type for both our cross-validation schemes for XGBoost, our best performing learner (Table 2). We observe a higher accuracy, precision, and recall on the FTL mistake than on the KOT and KK mistake types. We can think of two hypotheses as to why this is the case. First, based on preliminary research and manual examination of the data, we hypothesize that the Kinect tracks the upper body more accurately than the lower body. This would naturally explain why we can accurately detect *FTL*, which is a mistake related to the upper body, and not *KOT* nor *KK*, which are mistakes related to the lower body. However, further research is

needed to confirm this hypothesis. A second hypothesis is that our representation is imperfect in that it contains the information necessary to detect the *FTL* mistake, but lacks the necessary features to detect other mistakes. For example, *KK* is a type of mistake which will show almost no notable difference in angles of interconnected joint triplets, as the geometrical angle between the left and right *(hip, knee, ankle)* joint triplets will be unaffected. We conclude that the accuracy of AMIE depends on the type of mistake that was made during an exercise. To discern the exact reason as to why this is the case, further research is needed.

Table 2. Confusion matrices for detecting the type of mistake for (a) leave-one-set-out cross-validation and (b) leave-one-subject-out cross-validation.

Predicted / Actual	None	KOT	KK	FTL
None	816	30	19	30
KOT	161	10	20	5
KK	110	26	52	6
FTL	40	8	8	449

(a)

Predicted / Actual	None	KOT	KK	FTL
None	724	56	66	49
KOT	121	38	28	9
KK	75	10	101	8
FTL	37	3	6	459

(b)

5 Related Work

Previous work on the topic of home monitoring systems for rehabilitation exercises can be roughly divided in three categories: (1) work that qualitatively evaluates whether patients are willing to use a home monitoring system and what their expectations are of such a system [7,16]; (2) work that investigates the readiness and accuracy of several tracking systems to be used in a home monitoring system [10,21,26]; and (3) work that investigates home monitoring systems that can provide feedback using tracking data [1,6,10,15,25,26,29,30].

One of the hypotheses in our paper relevant to the second category is that the Kinect tracking system is not accurate enough to detect lower body mistakes. Pfister et al. [21] and Tang et al. [26] partially confirm this hypothesis by comparing the tracking capabilities of the Kinect to that of a Vicon camera system, which is considered to be the gold standard for tracking movements of the human body in biomechanics research. This hypothesis also explains why a large portion of the related work that incorporates the Kinect in a home monitoring systems focuses on upper body exercises [6,7,16,25,26].

Each paper in the third category contains one or more of three contributions: (a) describing the model in technical depth, (b) describing the used data set and experimental setup, and (c) outlining a clear vision on how the system should be implemented in practice. Typically, papers in this category contain either (a, c)

or (b, c), but rarely (a, b, c). We consider our paper to be an (a, b) paper. We did not outline a vision on how the system should be implemented in practice both for brevity and due to the fact that our paper is mostly a feasibility study.

Examples of (a, c) work are Anton et al. who introduce KiReS [1], Tang et al. who showcase Phyio@Home [26], and Zhao et al. who introduce a rule-based approach for real-time exercise quality assessment and feedback [29,30].

An example of (b, c) work is Komatireddy et al. who introduce the VERA system [15]. In terms of experimental setup, it is the most similar work to our paper; they collected data of ten healthy subjects within age 18–36 and asked them to perform ten correct repetitions of four different exercises (sitting knee extension, standing knee flexion, deep lunge, and squat). However, they provide no description on how the correctness of an exercise is determined and do not discuss the accuracy of the system compared to a physiotherapist in-depth.

Su et al. [25] wrote one of the few (a, b, c) papers. They introduce a distance-based approach to provide feedback on rehabilitation exercises using previous recordings. A physiotherapist first recorded correct executions of the exercises together with the patient. Feedback was then provided on new exercise executions at home using the distance to those reference executions. The task they consider is simpler than the one addressed in this paper however, because they evaluate their approach on shoulder exercises, which exhibit less complex movement patterns than the exercises we consider and are more accurately tracked by the Kinect. They also construct a per-subject model, which is easier than constructing a global model that can generalize over unseen subjects. A final note is that we could not find any information on the size of the employed test and training data, so it is unknown how reliable the estimated accuracy of their approach is.

6 Conclusion

We presented AMIE, a machine learning approach for automatically monitoring the execution of commonly used rehabilitation exercises using the Kinect, a low-cost and portable 3D-camera system. This paper contributes with respect to existing work by being one of the first to comprehensively detail the collected data set, describe the used classification system in depth, report quantitative results about our performance, and publicly release both the collected data set and used software tools.

We evaluated AMIE on a data set of ten test subjects who each performed six sets of ten repetitions of three commonly used rehabilitation exercises (i.e., squat, forward lunge and side lunge). AMIE detects the type of exercise with 99% accuracy and the type of mistake that was made with 73% accuracy. It does this almost in real-time. An important limitation of AMIE is that it can accurately detect movement mistakes of the upper body, but struggles with movement mistakes related to the lower body. We hypothesize that some non-trivial technical improvements (i.e., more accurate extraction of stick figures from depth frames and a better representation of our data) are necessary to solve the remainder of our task and implement the system in practice.

Acknowledgements. Tom Decroos is supported by the Research Foundation-Flanders (FWO-Vlaanderen). Kurt Schütte and Benedicte Vanwanseele are partially supported by the KU Leuven Research Fund (C22/15/015) and imec.icon research funding. Tim Op De Beéck and Jesse Davis are partially supported by the KU Leuven Research Fund (C22/15/015, C32/17/036).

References

1. Antón, D., Goñi, A., Illarramendi, A., Torres-Unda, J.J., Seco, J.: KiRes: a kinect-based telerehabilitation system. In: 2013 IEEE 15th International Conference on e-Health Networking, Applications & Services (Healthcom), pp. 444–448 (2013)
2. Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E.: The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Min. Knowl. Discov. **31**(3), 606–660 (2017)
3. Borgers, J., Vos, S., Scheerder, J.: Belgium (Flanders). In: Running Across Europe, pp. 28–58. Palgrave Macmillan, London (2015)
4. Campbell, R., Evans, M., Tucker, M., Quilty, B., Dieppe, P., Donovan, J.: Why don't patients do their exercises? Understanding non-compliance with physiotherapy in patients with osteoarthritis of the knee. J. Epidemiol. Commun. Health **55**(2), 132–138 (2001)
5. Chan, D.K., Lonsdale, C., Ho, P.Y., Yung, P.S., Chan, K.M.: Patient motivation and adherence to postsurgery rehabilitation exercise recommendations: the influence of physiotherapists' autonomy-supportive behaviors. Arch. Phys. Med. Rehabil. **90**(12), 1977–1982 (2009)
6. Chang, C.Y., et al.: Towards pervasive physical rehabilitation using Microsoft Kinect. In: 2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth), pp. 159–162. IEEE (2012)
7. Chang, Y.J., Chen, S.F., Huang, J.D.: A kinect-based system for physical rehabilitation: a pilot study for young adults with motor disabilities. Res. Dev. Disabil. **32**(6), 2566–2570 (2011)
8. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM (2016)
9. Council, Physical Activity: 2018 participation report: the Physical Activity Council's annual study tracking sports, fitness, and recreation participation in the US (2017)
10. Fernández-Baena, A., Susín, A., Lligadas, X.: Biomechanical validation of upper-body and lower-body joint movements of kinect motion capture data for rehabilitation treatments. In: 2012 4th International Conference on Intelligent Networking and Collaborative Systems (INCoS), pp. 656–661. IEEE (2012)
11. Foulds, J., Frank, E.: A review of multi-instance learning assumptions. Knowl. Eng. Rev. **25**(1), 1–25 (2010)
12. Stanford Children's Health: Sports Injury Statistics (2010). http://www.stanfordchildrens.org/en/topic/default?id=sports-injury-statistics-90-P02787
13. Hootman, J.M., Dick, R., Agel, J.: Epidemiology of collegiate injuries for 15 sports: summary and recommendations for injury prevention initiatives. J. Athl. Train. **42**(2), 311–319 (2007)
14. Knight, E., Werstine, R.J., Rasmussen-Pennington, D.M., Fitzsimmons, D., Petrella, R.J.: Physical therapy 2.0: leveraging social media to engage patients in rehabilitation and health promotion. Phys. Ther. **95**(3), 389–396 (2015)

15. Komatireddy, R., Chokshi, A., Basnett, J., Casale, M., Goble, D., Shubert, T.: Quality and quantity of rehabilitation exercises delivered by a 3D motion controlled camera. Int. J. Phys. Med. Rehabil. **2**(4) (2014)
16. Lange, B., Chang, C.Y., Suma, E., Newman, B., Rizzo, A.S., Bolas, M.: Development and evaluation of low cost game-based balance rehabilitation tool using the Microsoft Kinect sensor. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, pp. 1831–1834. IEEE (2011)
17. Levac, D.E., Miller, P.A.: Integrating virtual reality video games into practice: clinicians' experiences. Physiother. Theory Pract. **29**(7), 504–512 (2013)
18. de Loes, M., Dahlstedt, L., Thomee, R.: A 7-year study on risks and costs of knee injuries in male and female youth participants in 12 sports. Scand. J. Med. Sci. Sports **10**(2), 90–97 (2000)
19. Palazzo, C., et al.: Barriers to home-based exercise program adherence with chronic low back pain: patient expectations regarding new technologies. Ann. Phys. Rehabil. Med. **59**(2), 107–113 (2016)
20. Pedregosa, F., et al.: Scikit-learn: machine learning in python. Journal of machine learning research **12**(Oct)
21. Pfister, A., West, A.M., Bronner, S., Noah, J.A.: Comparative abilities of Microsoft Kinect and Vicon 3D motion capture for gait analysis. J. Med. Eng. Technol. **38**(5), 274–280 (2014)
22. Pisters, M.F., et al.: Long-term effectiveness of exercise therapy in patients with osteoarthritis of the hip or knee: a systematic review. Arthritis Care Res. **57**(7), 1245–1253 (2007)
23. Ray, S., Scott, S., Blockeel, H.: Multi-instance learning. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning, pp. 701–710. Springer, Boston (2011). https://doi.org/10.1007/978-0-387-30164-8_569
24. Sheu, Y., Chen, L.H., Hedegaard, H.: Sports-and recreation-related injury episodes in the united states, 2011–2014. Natl. Health Stat. Rep. (99), 1–12 (2016)
25. Su, C.J., Chiang, C.Y., Huang, J.Y.: Kinect-enabled home-based rehabilitation system using dynamic time warping and fuzzy logic. Appl. Soft Comput. **22**, 652–666 (2014)
26. Tang, R., Yang, X.D., Bateman, S., Jorge, J., Tang, A.: Physio@home: exploring visual guidance and feedback techniques for physiotherapy exercises. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 4123–4132. ACM (2015)
27. Taylor, M.J., McCormick, D., Shawis, T., Impson, R., Griffin, M.: Activity-promoting gaming systems in exercise and rehabilitation. J. Rehabil. Res. Dev. **48**(10), 1171–1186 (2011)
28. Wojcicki, J.M., Heyman, M.B.: Let's move - childhood obesity prevention from pregnancy and infancy onward. N. Engl. J. Med. **362**(16), 1457–1459 (2010)
29. Zhao, W.: On automatic assessment of rehabilitation exercises with realtime feedback. In: 2016 IEEE International Conference on Electro Information Technology (EIT), pp. 0376–0381. IEEE (2016)
30. Zhao, W., Feng, H., Lun, R., Espy, D.D., Reinthal, M.A.: A kinect-based rehabilitation exercise monitoring and guidance system. In: 2014 5th IEEE International Conference on Software Engineering and Service Science (ICSESS), pp. 762–765. IEEE (2014)