

# A User-Centered Design Approach to Physical Motion Coaching Systems for Pervasive Health

Norimichi Ukita<sup>1</sup>(✉), Daniel Kaulen<sup>2</sup>, and Carsten Röcker<sup>2</sup>

<sup>1</sup> Nara Institute of Science and Technology, Takayama, Ikoma, Nara 8916-5, Japan  
ukita@is.naist.jp

<sup>2</sup> RWTH Aachen University, Campus-Boulevard 57, 52074 Aachen, Germany  
roecker@comm.rwth-aachen.de

**Abstract.** Our goal is to develop a system for coaching human motions (e.g., for rehabilitation and daily health maintenance). This paper focuses on how to coach a user so that his/her motion gets closer to the good template of a target motion. It is important to efficiently advise the user to emulate the crucial features that define the good template. The proposed system (1) automatically mines the crucial features of any kind of motion from a set of motion features and (2) gives the user feedback about how to modify the motion through an intuitive interface. The crucial features are mined by feature sparsification through binary classification between the samples of good and other motions. An interface for motion coaching is designed to give feedback via different channels (e.g., visually, aurally), depending on the type of error. To use the total system, all the user must do is just move and then get feedback on the motion. Following experimental results, open problems for future work are discussed.

**Keywords:** Motion coaching · Error feedback · Physical rehabilitation

## 1 Introduction

### 1.1 Background

The number of people suffering from chronic diseases is constantly rising [1–3]. Today, more than three quarters of the elderly population are suffering from chronic diseases, independent of the economic, social, and cultural background [4]. However, not only the prevalence of chronic illnesses increases with age but also the likeliness of suffering from physical as well as mental disabilities. Statistical data from Great Britain [5] shows that around half of all disabled persons are 65 years or older.

A serious problem closely connected with declining physical abilities is an increased risk of falls. Statistics of the World Health Organization [6] show that approximately one third of the people over 65 years and half of the people over 80 years of age fall each year. Similar data is reported by Nehmer et al. [7]. Around 20% to 30% of the falls lead to serious injuries with long-term consequences for the patients [8]. Statistical data from the UK [9] shows that falls are the major cause for disability in the age group of people over 75 and a leading cause of mortality due to injury. The most common serious injuries related

to falls in older people are hip fractures, which result in annual costs of over 2 billion Euros for England alone [10].

The demographic change, which can be observed in most industrialized countries around the globe, does not only lead to an increased number of elderly people but also contributes to a continuous decline of the working population. For example, it is expected that the working force in Europe will decrease by 48 million people until 2050 while the dependency ratio is expected to double, reaching 51 % in the same time [11]. Consequently, the ratio between the working population and older citizens above 65 will shrink from currently 4:1 to only 2:1 in the coming 40 years. This development will inevitably result in a reduction of the number of people who can provide care to older and disabled people [8]. Together with the financial constraints that most are currently facing, it will become increasingly difficult to find enough caregivers for the growing number of elderly people [12].

In this context, pervasive homecare environments are often cited as a promising solution for providing automated and personalized healthcare solutions for a growing number of elderly people [7, 10, 12]. Pervasive healthcare environments are usually equipped with different types of sensors for automated data capturing as well as different types of output devices including large screens [13–15], mobile devices [16, 17], and ambient displays [18, 19]. Over the last decade, several prototype systems have been developed (e.g., [2–5, 8, 11]), which demonstrate the potential of such environments for individually supporting different user groups [6, 20].

Within this paper, we describe the development of an automatic motion coaching system which makes use of typical input and output technologies available in pervasive homecare environments in order to provide new user-centered training and rehabilitation concepts [9]. For easy-to-use coaching systems [1], it is important to efficiently advise a user to emulate the crucial features that define the good template. This is because many other features of the target motion might be varied among individuals, but those variations give less impacts on evaluating the target motion. The proposed method automatically mines the crucial features of any kind of motion. The crucial features are mined based on feature sparsification through binary classification between the samples of good and other motions. The following section provides a more detailed overview of the proposed system.

## 1.2 Our Approaches and Related Work

**Motion Measurement.** We aimed at developing a user-centered system for coaching human movement. For motion measurement in the laboratory stage, multi-camera systems, [21–23], allow us to acquire highly accurate results, but they are too expensive for realizing pervasive health systems. We have seen a tremendous improvement of commercial real-time motion tracking devices. Systems like Microsoft Kinect, Nintendo Wiimote, or PlayStation Move provide low-cost solutions for end-users in home environments. The proposed system utilizes an inexpensive depth-measurement sensor (i.e., Microsoft Kinect) in

order to get high-measurement accuracy without devices attached to the body for easy-to-use operation.

**Motion Coaching Systems.** During the last years, several motion coaching systems have been developed. Most systems focus on a special type of motion or exercise. This is due to the fact that there are tremendous differences between motions that have to be considered when analyzing motion data programmatically.

A review of several virtual environments for training in ball sports was introduced in [24]. They stressed that coaching and skill acquisition usually involve three distinct processes: conveying information (i.e., observational learning), structuring practice (i.e., contextual inference), and the nature and administration of feedback (i.e., feedback frequency, timing, and precision). Additionally, general possibilities when to provide feedback were identified. Concurrent feedback (during), terminal feedback (immediately following), or delayed feedback (some period after) can be used to assist the subject in correcting the motion.

One recent concurrent feedback approach was taken by Velloso et al. [25]. Another example for concurrent feedback was presented by Matsumoto et al. [26] who combined visual and haptic feedback. Even though their device greatly improved the performance, it was very awkward to perform the exercises with it due to its weight.

How to assist weightlifting training by tracking the exercises with a Kinect and using delayed feedback is proposed by Chatzitofis et al. [27]. However, there is still need for a human trainer to interpret those values in order to give feedback to the subject. The tennis instruction system developed by Takano et al. [28] also uses a delayed feedback approach but the focus is put on the process of observational learning. Due to the absence of any explicit feedback in [28], it is hard to determine how to actually correct the motion.

An example for terminal feedback can be found in [29] where the focus is put on the correct classification of motion errors while feedback is given immediately after the completion of the motion. However, this only allows the correction of previously known and trained error types.

To systematically analyze possible designs of motion coaching systems, the related work can be classified in a three-dimensional design space of multimodality [30]. The modality (visual, auditory, haptic) is chosen depending on the type of input that the computer or human needs to perceive or convey information.

A single system generally consists of multiple points in this design space (represented as a connected series of points). For example, the system developed by Chatzitofis et al. [27] can be controlled with mouse and keyboard (haptic input of control), visualizes performance metrics (visual output of data), and captures motion data by using the Kinect system (visual input of data).

In some cases, the differentiation between output of control and data is not unambiguous. Nevertheless, this can still be visualized. For example, in [25] the output of an arrow indicating the direction in which to move the left or right arm can be regarded as both, output of data and control. In the following, this type of visualization will be referred to as output of control.

## 2 Glossary

**Depth sensor** is an optical sensor that measures 3D distance from the sensor to 3D points in a scene. The measured results are obtained as a gray-scale image in which each pixel value represents the 3D distance. The examples of the depth image are shown in Fig. 1 (i.e., “Depth images” in the figure). There are several kinds of depth sensors, which are classified by a mechanism for measuring 3D distance.

Expensive but accurate sensors are based on Time-Of-Flight (TOF) measurement. A TOF sensor measures 3D distance by measuring the lapse of time after the sensor emits light and before the light returns to the sensor.

Some other depth sensors are based on triangulation. Unlike human eyes that observe a 3D point from different view points for triangulation (which is often called stereo vision), many triangulation-based depth sensors emit light and observe it from a different viewpoint for triangulation. This approach allows us to easily measure 3D distance because point correspondence is easy; in stereo vision, on the other hand, we must make a pixel correspondence between different views (i.e., different images) based on noisy image features so that pixels observing the same 3D point are paired.

Structured-light based sensors are also popular. These sensors emit a known spatial light pattern and observes it. Based on its deformation projected on a 3D surface in a scene, depth measurement can be achieved.

**Kinect** is a world-wide popular depth sensor developed by Microsoft. It can capture color and depth images simultaneously. Its depth measurement is based on the structured-light mechanism.

**Motion capture system** is used for obtaining the 3D human pose of a real person. A number of commercial products have been already developed, but all of them are still expensive. Several kinds of motion capture systems have been developed, namely optical systems with passive/active markers, inertial systems, mechanical systems, and magnetic systems.

**Support Vector Machine (SVM)** is a pattern classifier [31]. Any pattern is computationally expressed by a vector. In pattern classification, each pattern is attributed to a class (e.g., “good” or “bad”).

**3D Human Pose** (aka a 3D skeleton) is computationally represented by a set of 3D joint positions and links that connect physically-connected joints. Its examples are illustrated in Fig. 1 (i.e., “Pose sequence” in the figure).

## 3 State-of-the-Art

### 3.1 System Overview

Figure 1 illustrates the overview of the proposed system consisting of two steps.

An offline model-learning step is performed before users are coached by the system. In this step, two kinds of computational models are trained. For learning

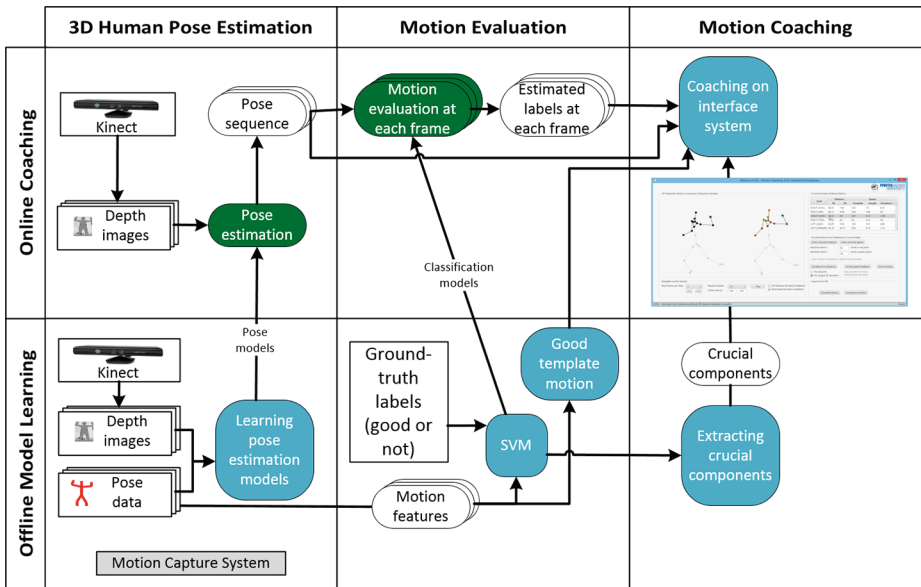


Fig. 1. Overview of the system.

the pose estimation model (i.e., “Pose models” in Fig. 1) that represents the relationship between human poses and features extracted from depth images, the samples of a target motion are captured by a synchronized Kinect and motion capture system (i.e., “Kinect” and “Motion capture system” of “Offline model learning” in Fig. 1). The pose classification model (“Classification models” in Fig. 1) is acquired by the Support Vector Machine [32] (“SVM” in Fig. 1) to evaluate whether the human pose at each frame is good or not.<sup>1</sup> In addition, the crucial features of the target motion (i.e., “Crucial components” in Fig. 1) are mined by a sparse coding regularization in the SVM.

In an online coaching step, with the model learned beforehand, the system observes the motion of a user with a Kinect camera (i.e., “Kinect” of “Online coaching” in Fig. 1), estimates the human pose at every frame (i.e., “Pose estimation” in Fig. 1), evaluates whether or not each pose is required to be modified (i.e., “Motion evaluation at each frame” in Fig. 1), and coaches the user. In the online coaching step, the three modules interact with a user as follows:

**3D human pose estimation:** A 3D human pose at each frame is estimated from a depth image captured by a Kinect. The estimation method is based on [33,34]. The accuracy of the pose estimation is improved by using real pose data captured by the motion capture system instead of synthesized CG data employed in [33,34].

<sup>1</sup> We assume that a target motion can be classified into good and other motions. For example, any motion in rehabilitation should be as correct (i.e., good) as possible.

**Motion evaluation:** The user’s pose is evaluated by the SVM whether it is good or not. If the pose is not good, it must be modified so that it gets closer to a good template. Before evaluation, the pose sequence of the user is synchronized with that of the template by dynamic time warping [35].

**Motion coaching:** At each subsequence (i.e., several sequential frames) that must be modified, the interface system [36] gives feedback to the user. Note that there might be a number of differences between the user’s motion and the good template motion and that it is actually impossible to understand all of them simultaneously. The proposed interface system gives feedbacks one by one, depending on their priority. More crucial features are given first. The priority of a feature is determined by how crucial the feature is for a well done execution of the good template motion.

### 3.2 3D Human Pose Estimation

The estimation method is based on [33, 34]. In this previous method, all training data (i.e., “Depth images” and “Pose data” in Fig. 1) are generated from simulation computer graphics data. This approach is useful for estimating arbitrary human poses for gaming purposes because

- it is difficult to collect the synchronized human pose and depth data of a large variety of arbitrary human poses, and
- even if the pose estimation error is relatively large due to modeling errors of a variety of human poses, it might still be acceptable for gaming purposes.

In contrast to pose estimation for gaming purposes, for motion coaching it should be more accurate. In particular, accurate pose estimation is required for rehabilitation purposes.

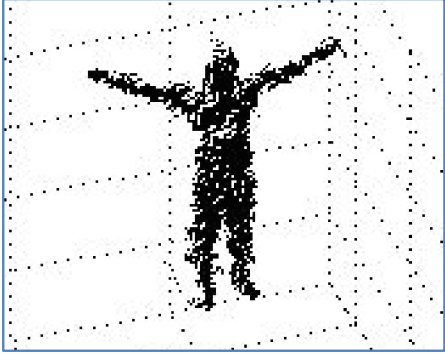
In the proposed system, accuracy in pose estimation is improved by using real observation data of human motions. The real depth images and human data are captured by Kinect and a motion capture system. The left and right images in Fig. 2 show a 3D point cloud computed from a captured depth image and a skeletal human pose, respectively.

From a technical point of view, spatial alignment and temporal synchronization between the point cloud and the pose are required.

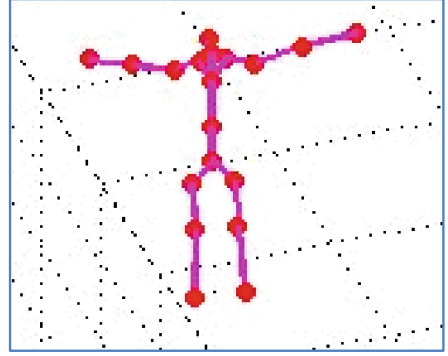
**Temporal Synchronization.** Since a moving human body is captured by two independent sensors, their captured data must be synchronized; a depth image and pose data in the same frame of an image sequence and a pose sequence must be captured at the same moment.

Unfortunately, Kinect does not have a hardware synchronization mechanism. Instead, in the proposed system a software synchronization between those two data sets is established.

For this synchronization, a predefined motion is performed by a subject. This predefined motion is required to have a key frame that can be easily identified in both depth image and pose sequences. The depth image and pose sequences are



Depth data extracted from a depth image captured by Kinect



Human pose data (skeleton) obtained by a motion capture system, IGS-190

**Fig. 2.** Real depth and pose data, necessary to create a 3D pose model.

synchronized so that each of their key frames is a first frame in each sequence. After the first frame,  $f_I$ -th frame of the image sequence is temporally aligned with  $f_P = \frac{F_P}{F_I}(f-1)+1$ -th frame, where  $F_P$  and  $F_I$  denote the frames-per-second of the motion capture system and Kinect, respectively. In the experiments shown in this paper, the subjects had to raise their right arm so that the key frame where the hand was located in the highest position could be identified.

**Spatial Alignment.** Kinect and the motion capture system have their own coordinate systems. They must be aligned in order to completely overlap the 3D point cloud and the pose of a subject.

Assume that the temporal synchronization is established. Spatial alignment is achieved by translating and rotating the coordinate system of one of the two sensors (i.e., in our experiments the motion capture system) so that the 3D positions of several key points coincide with each other between the two coordinate systems. Since the number of unknown parameters is 6 (i.e., 3 degrees of freedom in translation and 3 degrees of freedom in rotation), at least 2 pairs of corresponding points between the point cloud and the human pose are needed for estimating those unknown parameters; each pair gives us 3 equations (i.e.,  $x$ ,  $y$ , and  $z$  matching).

The following Eq. (1) expresses the translation and rotation of a 3D point  $\mathbf{M}$ :

$$\mathbf{M}' = \mathbf{T} + \mathbf{R}\mathbf{M}, \quad (1)$$

$$\mathbf{T} = (t_x, t_y, t_z)^T, \quad (2)$$

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

Here,  $t_x$ ,  $t_y$ ,  $t_z$ ,  $\alpha$ ,  $\beta$ , and  $\gamma$  are 6 unknown parameters. Given a 3D position  $\mathbf{M}_P$  of the human pose, these 6 parameters are optimized so that  $\mathbf{M}'_P$  is equal to the corresponding 3D position  $\mathbf{M}_I$  of the point cloud. Note that  $\mathbf{M}_P$  and  $\mathbf{M}_I$

are extracted from the same 3D point captured by the motion capture system and Kinect, respectively. In reality,  $\mathbf{M}'_P$  might not be equal to  $\mathbf{M}_I$  due to noise. Therefore, the 6 parameters are optimized so that  $\|\mathbf{M}'_P - \mathbf{M}_I\|$  gets close to zero.

Since the above-mentioned problem is a non-linear optimization, we have a variety of options for its solution. The LevenbergMarquardt algorithm was used in our experiments. If three or more corresponding points are available, parameter estimation can be robust to noise.

**Random Forest Regression for 3D Pose Estimation.** Given two spatially-aligned and temporally-synchronized sequences (i.e., 3D point cloud and human pose sequences), 3D pose estimation can be achieved in the exact same way with [34]. All frames in the sequences are used for model learning.

With the model learned, we can obtain the 3D human pose at each frame only from a depth image captured by the Kinect. The sequence of the estimated human poses is employed for motion evaluation described in the following section.

### 3.3 Mining Crucial Features for Motion Evaluation

**Mining Crucial Features via Sparse Coding.** For evaluating the motion of a user (i.e., classifying the motion to good or other motions), the SVM is used in the proposed system. This classification is performed with a number of features that represent the 3D pose and the motion of a human body. Since (1) the system should be applicable to any kind of motion and (2) we do not know which features of a target motion are crucial for defining the target motion, it is better to exhaustively use all features that possibly represent a body motion. In experiments, the concatenation of the following components was used as a 621D feature vector which consists of:

- 3D positions of all joints ( $3\text{D} \times 18 \text{ joints} = 54\text{D}$ )
- 3D velocities of all joints ( $3\text{D} \times 18 \text{ joints} = 54\text{D}$ )
- 3D accelerations of all joints ( $3\text{D} \times 18 \text{ joints} = 54\text{D}$ )
- 3D displacement between any pairs of joints ( $3\text{D} \times 153 = 459\text{D}$ ).

From these 621 features, the proposed method automatically mines which body parts and/or motions are crucial for improving the movement of a user. This mining is achieved by the sparse coding regularization in the SVM, as proposed in [37]. In classification, the inner product of the feature vectors of a test pose (denoted by  $\mathbf{v}$ ) and the weight vector  $\mathbf{w}$  is computed. If the inner product is above/below 0, the test pose is regarded as a positive/negative class (e.g., good/others). Therefore, components with a larger absolute value in  $\mathbf{w}$  correspond to crucial features that have a large impact on the inner product. In learning the SVM, the  $l_1$ -regularized logistic regression [37] is employed so that the gap between larger and smaller absolute values of  $\mathbf{w}$  gets much greater.

This sparsification can be regarded as dimensionality reduction because the dimensions with smaller values can be neglected. For dimensionality reduction, many other techniques have been proposed (e.g., PCA, LLE [38], Isomap [39], GPLVM [40]). Those techniques, however, cannot provide a user intuitive



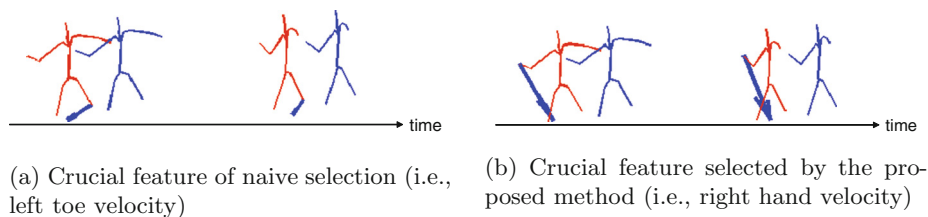
feedbacks for understanding how to modify the motion. This is because these techniques project a vector from a high-dimensional space to a low-dimensional space defined by an arbitrary subspace in the high-dimensional space. That is, each axis in the low-dimensional space might correspond to multiple axes in the original high-dimensional space. As a result, even if a motion feature corresponding to only one axis in a subspace obtained by PCA or similar techniques is selected, a user might be required to move the body as follows: “you should move the right hand, the right elbow, the left toe, and the hip so that ...”. On the other hand, in the proposed method, only one motion feature (e.g., “the right hand” or “the right elbow”) is selected from the low-dimensional space generated by the sparse coding regularization.

**Experiments of Feature Mining.** Experiments were conducted with baseball pitching motions<sup>2</sup> captured from 34 people; 13 good (i.e., expert) players and 21 beginners. From these 34 people, 445 sequences were captured in total. Both pose estimation and classification models were trained by the data of 33 people, and the data of the remaining one person was used for testing. Note that all 621 features were normalized.

The following two ways were tested for selecting crucial motion features:

- (a) **Naive selection:** The distance between features of a user’s pose and a good template is computed at each feature component (e.g., 3D position of the right hand); the distance at  $i$ -th feature is denoted by  $d_f$ . Features having the larger distance are regarded as crucial features.
- (b) **Selection with the sparsification:**  $d_f$  is multiplied with the weight of  $f$ -th feature (i.e.,  $f$ -th component of  $\mathbf{w}$ ). Features having the larger product are regarded as crucial features.

Motions selected by the above two criteria, (a) and (b), were checked by expert players. In examples shown in Fig. 3, naive selection recommended the left toe velocity as the most crucial motion (i.e., (a) in Fig. 3), while the right hand was selected by the proposed method (i.e., (b) in Fig. 3). It is natural that the motion of the hand holding a ball is more important for pitching. The experts also validated the selection of the proposed method.



**Fig. 3.** Visual feedbacks illustrating the difference between the user’s motion (shown with red) and the good template (shown with blue) in pitching motions (Colour figure online).

<sup>2</sup> To validate the system, a sport motion is a good example because its exercise is important for skill proficiency of beginners as well as rehabilitation of experts.

**Experiments of Using Features for Motion Evaluation.** We also demonstrate the effectiveness of the sparsification in motion evaluation. The mean classification rate of all 445 sequences, each of which was evaluated by leave-one-out cross-validation, is computed. The means over all frames were 67% and 76% in (a) and (b), respectively. These results demonstrate the effectiveness of the sparsification also in motion classification. This effect is gained because a low-dimensional feature space allows us to improve the generalizing capability of classifiers such as the SVM, as described in [41].

### 3.4 Motion Coaching Interface

Just a simple visualization of a motion difference (e.g., Fig. 3) might not be intuitive for motion coaching, depending on the type of motion error. This section discusses what kind of feedback is appropriate for each type of motion error.

**Interface Design.** To combine the ideas of motion errors and different types of motion feedback, a prototype system was implemented that enables first experiments with some of the proposed feedback types.

JavaFX was used as an underlying framework since it allows fast creation of user interfaces with JavaFX Scene Builder and provides built-in support for animations and charts. For the user to be able to concentrate on the visualization, the system takes two synchronized motion sequence files that contain information about joint positions at each point in time as input. Synchronized in this context means that frame number  $i$  in the template motion corresponds with frame number  $i$  in the comparison motion. Figure 4 shows a screenshot of the system. In this interface, joints that are not relevant for a special motion can be de-selected manually.

**Motion Errors and Feedback Types.** The first step when thinking about how to provide motion error feedback is to become aware of different types of

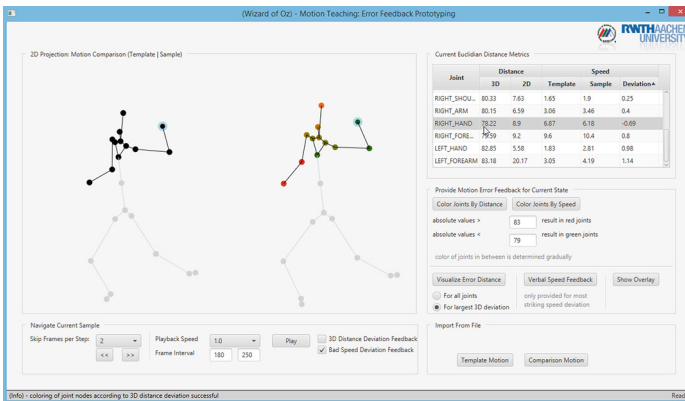


Fig. 4. Screenshot of the motion coaching system.

motion errors (i.e., deviation between a template and comparison motion) that need to be addressed. To that extent, it is obvious to differentiate between the spatial and temporal dimension.

When just considering the spatial dimension, there are three main types of motion errors that can occur. First, the absolute position of a joint can be wrong. When only the spatial collocation of several joints is important, the relative position of them should be taken into account instead. For example, a clapping motion can be defined only by the spatial relationship between the palms (i.e., the palms touch each other or not). The last main error type is a wrong angle between neighboring joints. Naturally, the angle is influenced by the actual positions of the joints, but it is expected that a different type of visualization is required depending on whether the focus is put on the angle or the absolute joint position.

In a next step, several general ways to provide feedback by using different modalities were elaborated.

The most natural but technically the most complex way when using the visual channel is to either extract only the human body or to use the complete real scene and overlay it with visual feedback (e.g., colored overlay of body parts depending on the distance error). The natural scene reduces the cognitive load for the subject as the mapping between the real world and the visualization is trivial. Displaying the human body as a skeleton makes this mapping a bit harder but allows to put the focus on the motion itself. To compare a template with a comparison motion, the abstracted skeletons can be visualized side by side or in an overlaid manner, as shown in Fig. 5. It is expected that the overlaid view is mainly applicable when trying to correct very small motion errors. At a higher abstraction level, performance metrics such as speed or distance deviation per joint or body part can be displayed textually or graphically (i.e., with the aid of charts). There is, however, no information on how to correct the motion and the subjects need to interpret those values to improve their motion. To overcome this weakness, it is desirable to be able to visualize instructions (i.e., visual output of control) that guide users in correcting their motion. Two possible



**Fig. 5.** Visualization of two skeletons, one captured online and one of a template motion. Left: side-by-side comparison. Right: overlay.

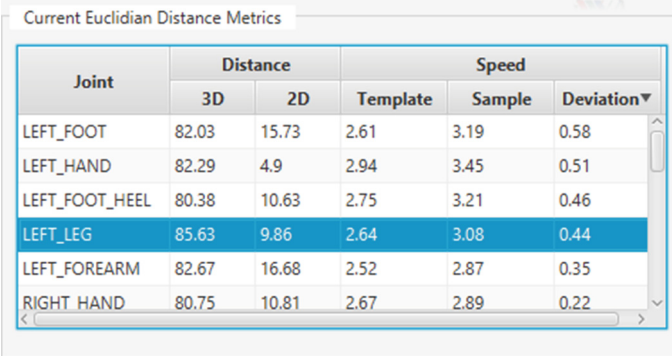
approaches are simple textual instructions [42] or graphical instructions such as arrows indicating the direction in which the motion should be corrected [25].

Audio feedback can be used in several ways to give motion error feedback. Spoken instructions (i.e., auditory output of control) are one possible way to which most people are already used to from real training situations. Note that the bandwidth of the auditory channel is much lower than the one of the visual channel and therefore not much information can be provided at the same time. Nevertheless, the audio feedback has the big advantage that it easily catches human attention and users do not have to look in a special direction (e.g., for observing a screen). In terms of auditory output of data, different parameters of sound (i.e., frequency, tone, volume) can be modified to represent special motion errors. A first step in this direction was taken by Takahata et al. [43] in a karate training scenario.

Another important point of research is the question of how to motivate people to use a motion coaching system. As it is commonly accepted that the use of multiple modalities increases learning performance (see [44], for example), a motion coaching system should aim at addressing multiple senses. Therefore, several of the ideas above are combined in the proposed interface.

The use of haptic output devices is not treated as applicable for a motion coaching system used to teach a wide range of different exercises due to two main reasons: First, there is no reliable and generic way to translate instructions into haptic patterns (see [45], for example). Second, specially adapted hardware is required to provide appropriate haptic feedback, which is then often considered disturbing [26].

**Multimodal Feedback Types Visual Output of Data 1 – Metrics (Textual).** The performance metrics illustrated in Fig. 6 provide basic information such as 3D and 2D distance deviations per joint and a comparison of the template and sample speed per joint. Due to the perspective projection of the real-world 3D coordinates to the joint positions in the visualized 2D skeleton on



Joint	Distance		Speed		
	3D	2D	Template	Sample	Deviation▼
LEFT_FOOT	82.03	15.73	2.61	3.19	0.58
LEFT_HAND	82.29	4.9	2.94	3.45	0.51
LEFT_FOOT_HEEL	80.38	10.63	2.75	3.21	0.46
LEFT_LEG	85.63	9.86	2.64	3.08	0.44
LEFT_FOREARM	82.67	16.68	2.52	2.87	0.35
RIGHT HAND	80.75	10.81	2.67	2.89	0.22

**Fig. 6.** Distance and speed metrics for a single pair of frames for currently loaded motion sequences.

the screen, it may occur that there are large 3D deviations that are not recognizable in the skeleton representation. The data helps to get an understanding of this relation and allows for a very detailed motion analysis. Nevertheless, this high precision is not necessarily needed for a motion coaching scenario and a subject may only use this type for terminal or delayed feedback.

**Visual Output of Data 2 – Metrics (Graphical).** Charts are used to visualize distance and speed metrics over time. Multiple joints can be selected to be included in a single chart to compare the respective deviations. From a motion coaching perspective, this type of feedback is mainly suited for terminal or delayed feedback. Figures 7 and 8 visualize the deviations of the distance and the speed (between the template and comparison motion) of two different joints for a small frame interval, respectively. As real-world data is often subject to large fluctuations, values are smoothed for visualization purposes by calculating a weighted average for the k-step neighborhood (k between 5 and 10), see an

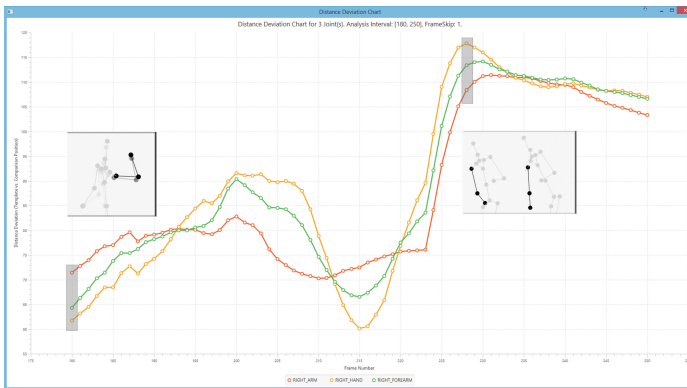


Fig. 7. Distance deviation chart for right forearm (selected series) and right hand.

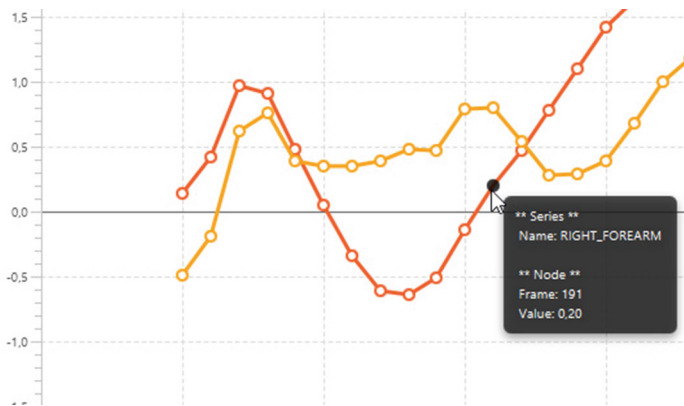


Fig. 8. Speed deviation chart for right forearm (selected series) and right hand.



Fig. 9. The effect of temporal smoothing. Upper: original data. Lower: smoothed data.

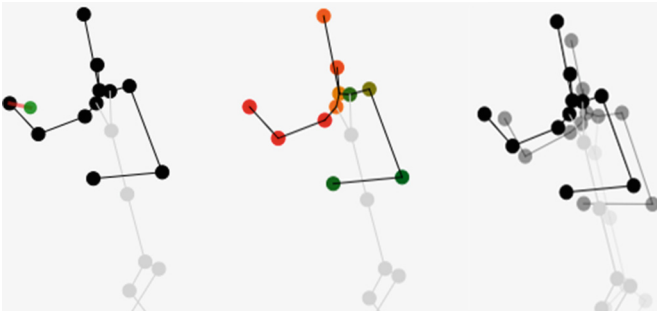


Fig. 10. Exemplary skeleton-based distance error visualizations (left: colored joint overlay, center: overlay of template and comparison skeleton, right: static result of animated joint moving to its correct position).

example in Fig. 9. While the original data was noisy (the upper graph in the figure), its smoothed graph is better for understanding the motion (the lower graph in the data).

**Visual Output of Data 3 – Colored Joint Overlay.** All joints with deviations larger than upper and lower thresholds, which are given manually, are respectively colored in red and green (applicable for speed and distance deviations). The coloring of joints with values in between those thresholds is determined gradually (i.e., vary from red over orange to green). An example can be found in Fig. 10 (left skeleton): the largest deviations occur for joints located in the right arm. This visualization approach can be used either for concurrent,

terminal, or delayed feedback and allows to easily determine joints with high deviations.

**Visual Output of Data 4 – Skeleton Overlay.** Visualizing the template and comparison skeleton in an overlaid manner (instead of side by side which is the default behavior of the proposed system) turned out to be only suitable to correct very small motion errors. Otherwise the mapping between the intended and actual joint position is not directly visible. Oftentimes, it is hard to differentiate between the two skeletons. To overcome this weakness, the opacity value of the template is lower than the one of the comparison skeleton (see Fig. 10, center).

**Visual Output of Control – Distance Error Animation.** So far, no direct information on how to correct the motion was given. The initial idea of Velloso et al. [25] that uses directed arrows to indicate how to correct the motion was adapted and replaced by an animated joint that moves to its correct position and thus gradually changes its color from red (wrong position) to green (correct target position is reached). Even though this is still a quite technical representation, this approach is considered to be more natural than using arrows (see Fig. 10, right). However, it is only applicable for terminal or delayed feedback.

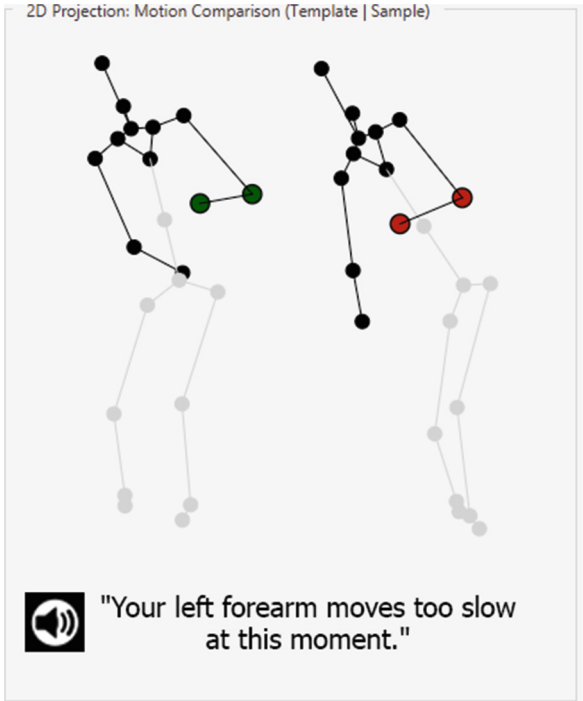
**Auditory Output of Control – Speed Feedback.** For the most striking speed deviation, a verbal feedback phrase is provided by using a text-to-speech library. However, even if humans are used to this type of auditory feedback, such a specific per-joint feedback is not applicable in practice. Therefore, several joints are clustered to body parts and feedback is provided accordingly (e.g., “Move your right arm faster” instead of “Move your right hand and elbow faster”). Auditory feedback in general is best suited for concurrent feedback.

**Combination of Visual and Auditory Output of Data.** As stressed in the previous section, per-joint speed feedback is regarded as too technical. In this approach that combines visual and auditory output, joints are clustered to body parts (by using the charts for analyzing deviation dependencies) and considered as a whole during motion error feedback. The animated illustration is embedded in a video playback of the motion sequences and supported by corresponding speech output, as illustrated in Fig. 11. Note that the coloring allows to easily determine the affected body part and the blinking speed of the highlighted joints depicts the type of speed deviation (too fast: fast blinking, too slow: slow blinking).

## 4 Open Problems

Our goal is to develop a user-centered physical motion coaching system which can be used for supporting private rehabilitation and training. For this coaching system, this chapter described (1) accurate 3D human pose estimation, (2) mining crucial motion features for efficient coach, and (3) intuitive motion-error feedback interface.

For 3D human pose estimation, its accuracy is improved by using real observation data (i.e., 3D point cloud captured by Kinect and 3D human pose data



**Fig. 11.** Example of embedded multimodal speed feedback in motion sequence playback (Note: text in the figure is provided by speech output and is not visualized).

obtained by a motion capture system) as training data. Since the two data are captured independently by different devices, spatial alignment and temporal synchronization are inevitable. While the system proposed in this chapter achieves these alignment and synchronization, the following questions remain open:

- Frame rate: Are the frame rates of the Kinect and the motion capture system completely identical?
- Drift: Does the 3D pose obtained by the motion capture system temporally drift at all?
- Frame(s) for temporal synchronization: Are the frames of the coordinate systems of the Kinect and the motion capture system appropriate for spatial alignment? Are there measurement errors in that frame? Would another frame be better suited for the alignment? Would it be better to use multiple frames for the alignment to cope with drift?

The crucial motions are mined by the sparse coding regularization during training of the SVM that classifies target motions into good or not. The weight vector of the SVM shows which features are crucial for classifying whether a user's motion is good or not. In particular, the sparse coding regularization allows us to enhance the difference between crucial and non-crucial features.



In reality, however, it is not so easy to extract only crucial features correctly from a huge number of possible features. While the ultimate goal of the system is to apply to any kind of motion fully automatically, it might be possible to reduce the possible features manually, based on knowledge of the motion so that only meaningful features (i.e., features that might be crucial) remain. Otherwise, for realizing a fully-automatic system, we might be able to reduce the possible features independently of the motion, based on the structure and general kinematics of a human body.

From a technical point of view, important issues for developing an intuitive motion-error feedback interface are not clear yet. In particular, how to automatically select a feedback type depending on the motion is an open problem. We need extensive user tests in order to address this.

## 5 Future Outlook

Future work for improving feature mining includes using knowledge of a human body (e.g., kinematics and joint structures) as heuristics, extensive user tests, and verification with many other kinds of motions. In terms of using the knowledge of a human body, it is expected that the knowledge is helpful for mining more discriminative features. Since this knowledge does not depend on the type of motion, usability of the system is not damaged.

For an intuitive interface system, different ways to provide motion error feedback were analyzed. The results from this first prototype can be used for an initial evaluation that may allow to exclude several feedback possibilities or reveal the need for analyzing others in more detail.

However, technology acceptance is a quite complex phenomenon and the success of a motion coaching system does not only depend on the interface alone. Final statements are only possible when a complete system has been developed and tested in detail. The development of such a system requires an interdisciplinary approach with scientific contributions from the fields of machine learning, computer vision, human-computer interaction, and psychology.

## References

1. Campana, F., Moreno, A., Riano, D., Laszlo, Z.: K4care: Knowledge-based home-care eservices for an ageing europe
2. Laleci, G.B., Dogac, A., Olduz, M., Tasyurt, I., Yuksel, M., Okcan, A.: Spahire: A multi-agent system for remote healthcare monitoring through computerized clinical guidelines. In: *Agent Technology and E-Health (2007)*
3. Villar, A., Federici, A., Annicchiarico, R.: K4care: Knowledge-based homecare eservices for an ageing europe. In: *Agent Technology and E-Health (2007)*
4. Vergados, D.J., Alevizos, A., Mariolis, A., Caragiozidis, M.: Intelligent services for assisting independent living of elderly people at home. In: *PETRA (2008)*
5. Population Censuses and Surveys Office: *General Household Survey 1994 (1995)*
6. World Health Organization: *Active Aging: A Policy Framework (2012)*

7. Nehmer, J., Becker, M., Karshmer, A.I., Lamm, R.: Living assistance systems: an ambient intelligence approach. In: ICSE, pp. 43–50 (2006)
8. de Ruyter, B.E.R., Pelgrim, E.: Ambient assisted-living research in carelab. *Interactions* **14**(4), 30–33 (2007)
9. Health Education Authority: Older People - Older People and Accidents, Fact Sheet 2 (1999)
10. Torgerson, D., Dolan, D.J.: The cost of treating osteoporotic fractures in the united kingdom female population
11. EU: The Demographic Future of Europe - From Challenge to Opportunity. Commission Communication (2006)
12. Adam, S., Mukasa, K.S., Breiner, K., Trapp, M.: An apartment-based metaphor for intuitive interaction with ambient assisted living applications. In: BCS HCI (1) (2008)
13. Röcker, C.: User-centered design of intelligent environments: Requirements for designing successful ambient assisted living systems. In: The Central European Conference of Information and Intelligent Systems (2013)
14. Röcker, C.: Smart medical services: a discussion of state-of-the-art approaches. *International Journal of Machine Learning and Computing*
15. Röcker, C., Ziefle, M.: Current approaches to ambient assisted living. In: The International Conference on Future Information Technology and Management Science and Engineering (2012)
16. Röcker, C., Maeder, A.: User-centered design of smart healthcare applications. *Electron. J. Health Inf.* **6**(2), 1–3 (2011)
17. Röcker, C.: Designing ambient assisted living applications: an overview of state-of-the-art implementation concepts. In: The International Conference on Information and Digital Engineering (2011)
18. Ziefle, M., Röcker, C., Holzinger, A.: Medical technology in smart homes: Exploring the user's perspective on privacy, intimacy and trust. In: COMPSAC Workshops, pp. 410–415 (2011)
19. Röcker, C., Ziefle, M., Holzinger, A.: Social inclusion in aal environments: Home automation and convenience services for elderly users. In: The International Conference on Artificial Intelligence (2011)
20. Röcker, C.: Intelligent environments as a promising solution for addressing current demographic changes. *Int. J. Innov. Manage. Technol.* **4**(1), 76–79 (2013)
21. Ukita, N., Hirai, M., Kidode, M.: Complex volume and pose tracking with probabilistic dynamical models and visual hull constraints. In: ICCV (2009)
22. Ukita, N., Kanade, T.: Gaussian process motion graph models for smooth transitions among multiple actions. *Comput. Vis. Image Underst.* **116**(4), 500–509 (2012)
23. Ukita, N.: Simultaneous particle tracking in multi-action motion models with synthesized paths. *Image Vis. Comput.* **31**(6–7), 448–459 (2013)
24. Miles, H.C., Pop, S., Watt, S.J., Lawrence, G.P., John, N.W.: A review of virtual environments for training in ball sports. *Comput. Graph.* **36**(6), 714–726 (2012)
25. Velloso, E., Bulling, A., Gellersen, H.: Motionma: motion modelling and analysis by demonstration. In: CHI (2013)
26. Matsumoto, M., Yano, H., Iwata, H.: Development of a motion teaching system using an immersive projection display and a haptic interface. In: WHC. (2007)
27. Chatzitofis, A., Vretos, N., Zarpalas, D., Daras, P.: Three-dimensional monitoring of weightlifting for computer assisted training. In: VRIC (2013)
28. Takano, K., Li, K.F., Johnson, M.G.: The design of a web-based multimedia sport instructional system. In: AINA Workshops (2011)

29. Chen, Y.J., Hung, Y.C.: Using real-time acceleration data for exercise movement training with a decision tree approach. *Expert Syst. Appl.* **37**(12), 7552–7556 (2010)
30. O’Sullivan, D., Igoe, T.: *Physical computing* (2004)
31. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
32. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. *ACM TIST* **2**(3), 27 (2011)
33. Shotton, J., Fitzgibbon, A.W., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: Cipolla, R., Battiato, S., Farinella, G.M. (eds.) *Machine Learning for Computer Vision. Studies in Computational Intelligence*, vol. 411, pp. 119–135. Springer, Heidelberg (2011)
34. Girshick, R.B., Shotton, J., Kohli, P., Criminisi, A., Fitzgibbon, A.W.: Efficient regression of general-activity human poses from depth images. In: *ICCV* (2011)
35. Myers, C.S., Rabiner, L.R.: Comparative study of several dynamic time-warping algorithms for connected-word recognition. *Bell Syst. Tech. J.* **60**(7), 1389–1409 (1981)
36. Ukita, N., Kaulen, D., Röcker, C.: Towards an automatic motion coaching system: feedback techniques for different types of motion errors. In: *International Conference on Physiological Computing Systems* (2014)
37. Li, L.J., Su, H., Xing, E.P., Li, F.F.: Object bank: a high-level image representation for scene classification and semantic feature sparsification. In: *NIPS* (2010)
38. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embeddin. *Science* **290**(5500), 2323–2326 (2000)
39. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)
40. Lawrence, N.D.: Probabilistic non-linear principal component analysis with gaussian process latent variable models. *J. Mach. Learn. Res.* **6**, 1783–1816 (2005)
41. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, Cambridge (2010)
42. Kelly, D., McDonald, J., Markham, C.: A system for teaching sign language using live gesture feedback. In: *FG* (2008)
43. Takahata, M., Shiraki, K., Sakane, Y., Takebayashi, Y.: Sound feedback for powerful karate training. In: *NIME* (2004)
44. Evans, C., Palacios, L.: Using audio to enhance learner feedback. In: *International Conference on Education and Management Technology* (2010)
45. Spelmezan, D., Borchers, J.: Real-time snowboard training system. In: *CHI Extended Abstracts* (2008)

## Reading

46. Ukita, N., Kaulen, D., Röcker, C.: Towards an automatic motion coaching system: feedback techniques for different types of motion errors. In: *International Conference on Physiological Computing* (2014)
47. Ukita, N., Kaulen, D., Röcker, C.: Mining crucial features for automatic rehabilitation coaching systems. In: *International Workshop on User-Centered Design of Pervasive Healthcare Applications* (2014)

48. Holzinger, A., Ziefle, M., Röcker, C.: Pervasive Health - State-of-the-Art and Beyond. Springer, London (2014)
49. Varshney, U.: Pervasive Healthcare Computing: EMR/EHR. Wireless and Health Monitoring. Springer, United States (2009)
50. Röcker, C., Ziefle, M.: Smart Healthcare Applications and Services: Developments and Practices. IGI Publishing, Niagara Falls (2011)
51. Bardram, J., Mihailidis, A., Wan, D.: Pervasive Computing in Healthcare. CRC Press Inc., Boca Raton (2006)
52. Röcker, C., Ziefle, M.: E-Health, Assistive Technologies and Applications for Assisted Living: Challenges and Solutions. IGI Publishing, Niagara Falls (2011)
53. Jähn, K., Nagel, E.: E-Health. Springer, Berlin (2003)
54. Ziefle, M., Röcker, C.: Human-Centered Design of E-Health Technologies: Concepts. Methods and Applications. IGI Publishing, Niagara Falls (2011)
55. Coronato, A., De Pietro, G.: Pervasive and Smart Technologies for Healthcare: Ubiquitous Methodologies and Tools. IGI Publishing, Niagara Falls (2011)