PostureCheck: Posture Modeling for Exercise Assessment Using the Microsoft Kinect

Elham Saraee, Saurabh Singh, Ajjen Joshi and Margrit Betke
Department of Computer Science, Boston University, Boston, MA, USA
esaraee@bu.edu, ssingh02@bu.edu, ajjendj@bu.edu, betke@bu.edu

Abstract—Evaluation of a person's posture while exercising is important in physical therapy. During a therapy session, a physical therapist or a monitoring system must assure that the person is performing an exercise correctly to achieve the desired therapeutic effect. In this work, we introduce a system called PostureCheck for exercise assessment in physical therapy. PostureCheck assesses the posture of a person who is exercising in front of the camera of the Microsoft Kinect. PostureCheck extracts unique features from the person's upper body during the exercise, and classifies the sequence of postures as correct or incorrect using Bayesian estimation and majority voting. If PostureCheck recognizes an incorrect posture, it specifies what the user can do to correct it. The result of our experiment shows that PostureCheck is capable of recognizing the incorrect postures in real time while the user is performing an exercise.

Keywords—Bayesian Estimation, Majority Voting, Microsoft Kinect, PostureCheck, Upper Body Physical Therapy.

I. Introduction

Recent advances in technology-assisted physical rehabilitation has motivated researchers to design systems and devices to help the patients through the therapy [1], [2], [3]. The use of the Kinect interface for physical therapy has been investigated previously in the literature. An extensive review [4] summarizes recent studies regarding the use of Kinect for physical rehabilitation and the validity of these experiments with regards to the Kinect's accuracy.

In this work, we designed a system that retrieves specified anatomical landmarks as input and classifies the user's postures as correct or incorrect in real time during an exercise. For this purpose, we use the Kinect v2.0 for tracking the user's skeleton and to record the X, Y and Z coordinates of the upper body of the user. If a posture is measured to be incorrect, the method provides information about correcting it. We implemented and tested this method in a system called PostureCheck that consists of the following steps:

- PostureCheck extracts unique features that represent the posture of the user's upper body during an exercise. These features include various angles of the shoulder and elbow, as well as the orientation of the frontal (coronal) plane of the upper body.
- PostureCheck uses Gaussian priors to model the correct posture and three incorrect postures that commonly occur during upper arm exercises. While the user is performing the exercise, PostureCheck labels the user's postures as "correct" any of three "incorrect" categories using the posterior probabilities of each model and based on majority on a sequence of labels to handle correlations between time signals of features and eliminate spurious results.

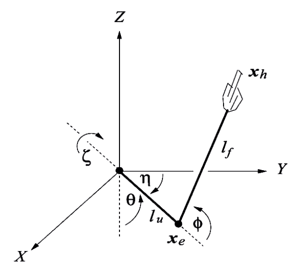


Fig. 1: Arm configuration defined by the three shoulder angles and the flexion angle at the joint elbow [5]

II. Method

A. Feature Extraction:

In order to analyze the user's posture, the PostureCheck system extracts the positions of six body points using the Kinect and then computes six features. We here focus on exercises performed with the right arm, and thus the landmarkd relate to the four degrees of freedom of the right shoulder and right elbow [5], as represented in Figure 1. Our method can be easily generalized to the left arm as well. The six extracted points are:

- Right hand (x_h, y_h, z_h)
- Right elbow (x_e, y_e, z_e)
- Right shoulder
- Left shoulder
- Neck
- Abdomen

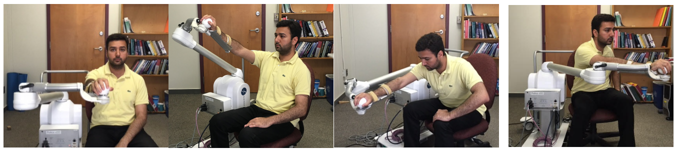
The six features computed from these points are:

1) Elevation angle of the upper arm
2) Azimuth angle of the upper arm
3) Humeral rotation angle of the upper arm
4) Elbow flexion/extension angle
5) Normal vector $\overrightarrow{n}$ that is perpendicular to the frontal (= coronal) body plane
6) Angle α between the vectors connecting the shoulders and the spine

The last two vectors that respectively connect neck to abdomen and the two shoulders are used as a representation spanning the frontal body plane. The normalized cross product and dot product of these two vectors are the two features that PostureCheck uses to describe the posture of the body.

B. Classification using a Gaussian Model

In order to predict whether the posture is correct or not, $p(y|x)$ is calculated using Bayes rules and the most probable label is selected, where x is the vector of the the obtained



(a) (b) (c) (d)

Fig. 2: Four postures considered in our dataset. Figure (a) shows the correct way of doing the exercise. Figures (b), (c), (d) show the class upward, downward and leaning forward respectively.

features and y is the label for the given posture. We model the distribution of each class of postures with a Gaussian prior. We construct our classifier from these probability models and apply maximum a posteriori (MAP) as the decision rule.
The probability that a given test posture X belongs to a specific class i is :

$$p(X|c_i) = \prod_{n=1}^{N} p(x_n|c_i) = \prod_{n=1}^{N} \mathcal{N}(x_n|\lambda_i), \tag{1}$$

where N is the number of features and λ denotes the parameters of the Gaussian model that is learned during the training. The correct class is the one that maximizes the posterior probability.

Post-Processing: The decision whether an exercise has been performed correctly or not does not depend on a single posture, but it is the result of a sequence of correlated postures that creates the exercise trajectory. In order to assign final labels for these correlated sequences, majority voting is applied to a sequence of labels.The size of the window used for majority voting is equal to the first minimum of the automutual information function [6] with in our setup corresponded to one fourth of the duration of the exercise. The delay between windows is equal to the half size of the window; thus each feature contributes equally to two labels in the final decision.

III. Experiment

In order to prepare our dataset, we asked eight individuals to perform an exercise in four different ways, in each case repeating the movement of the arm five to nine times. The exercise tested is moving the right arm in the transverse body plane from right to left and back to right again. The participants performed the exercise with the Proficio robotic arm while the Kinect camera was placed 1.5 meters in front of them. The use of robot imitates the scenario where the user is doing an exercise with an external help which might create occlusion. The four variations of the exercise include the **correct posture** where the back of the user is straight in the (yz) plane and the participant's arm moves in the transverse body plane (Figure 2(a)) and three **incorrect postures** which typically make the exercise easier for the user despite the risk of the injury that they might cause. 2(b) represents the posture where the user is leaning backward and the arm is pointing upward. We will refer to this category as ”posture upward”. Figure 2(c) depicts the user is leaning forward and his arm is pointing downward. We will refer to this category as ”posture downward”. In the third posture, instead of using the upper arm strength to move the arm, users tend to move their whole upper body, inserting force on the right side of their back, see Figure 2(d). We will refer to this category as ”posture moving shoulder”.

TABLE I: The confusion matrix for recognizing four possible postures during the exercising experiment. Each column corresponds to one of the prepared labels and each row corresponds to each hypothesized label for the sequence of postures.

	Correct	Downward	Upward	Shoulder moving
Hypothesized Correct	191	5	7	0
Hypothesized Downward	14	158	3	0
Hypothesized Upward	5	0	136	0
Hypothesized Shoulder Moving	0	0	0	75

We extracted features as explained in Section II-A. The total number of 21,097 features from our data set, 6,591 belonging to the correct posture, 5,929 features for the category downward, 6,000 features for category upward and 2,577 features for the last category, moving shoulder is obtained. We used the features extracted from four users as the training set, two users for cross validation and the other two for testing.

Result: A comparison of the class labels that PostureCheck computed for the exercises in our experiment and their gold standard labels shows that our system is capable of successfully distinguishing the four postures. For example, the correct positive rate (sensitivity) for the first category, correct posture, is 90.95%, and the true negative rate (specificity) for this category is $96,87\%$, the accuracy of detecting the correct posture class is 94.87%. Our full results are summarized in a confusion matrix in Table I. The off-diagonal numbers in the confusion matrix show that, in a few cases, false positive and false negative detections occur. Such occurrence may be due to the continuous nature of the problem – some exercise postures may not be clearly correct or incorrect, and so a ground truth boundary between posture classes is difficult to establish.

IV. Conclusions

In this paper, we introduced PostureCheck, a system that can distinguish correct and incorrect postures of a user exercising in front of the Kinect interface. Designing a telerehabilitation system that can engage the users more with their exercise as well as capable of monitoring their postures is our future goal to help anyone who needs physical therapy.

Acknowledgments: The work was supported in part by NSF (1337866).

References

[1] G. Kwakkel, B. J. Kollen, and H. I. Krebs, “Effects of robot-assisted therapy on upper limb recovery after stroke: a systematic review,” *Neurorehabilitation and neural repair*, 2007.

[2] C. Prakash, K. Gupta, A. Mittal, R. Kumar, and V. Laxmi, “Passive marker based optical system for gait kinematics for lower extremity,” *Procedia Computer Science*, vol. 45, pp. 176–185, 2015.

[3] N. C. Bejarano, S. Maggioni, L. De Rijcke, C. A. Cifuentes, and D. J. Reinkensmeyer, “Robot-assisted rehabilitation therapy: Recovery mechanisms and their implications for machine design,” in *Emerging Therapies in Neurorehabilitation II*. Springer, 2016, pp. 197–223.

[4] H. M. Hondori and M. Khademi, “A review on technical and clinical impact of Microsoft Kinect on physical therapy and rehabilitation,” *Journal of Medical Engineering*, vol. 2014, 2014, 16 pages, Article ID 846514.

[5] A. Biess, D. G. Liebermann, and T. Flash, “A computational model for redundant human three-dimensional pointing movements: Integration of independent spatial and temporal motor plans simplifies movement dynamics,” *The Journal of Neuroscience*, vol. 27, no. 48, pp. 13 045–13 064, 2007.

[6] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*. Cambridge University Press. Second Edition, 2003, cambridge Books Online. [Online]. Available: http://dx.doi.org/10.1017/CBO9780511755798