Data

Data_Correct: captured 117-dimensional data for the correct performance of the Deep Squat exercise.

Data_Incorrect: captured 117-dimensional data for the incorrect performance of the Deep Squat exercise.

Labels_Correct: corresponding quality scores for the correct movements obtained by the GMM loglikelihood of the data with reduced dimensionality by autoencoder.

Labels_Incorrect: corresponding quality scores for the incorrect movements obtained by the GMM loglikelihood of the data with reduced dimensionality by autoencoder neural network.

Autoencoder_Output_Correct: 4-dimensional data for the correct movements obtained by performing dimensionality reduction with autoencoder neural network.

Autoencoder_Output_Incorrect: 4-dimensional data for the incorrect movements obtained by performing dimensionality reduction with autoencoder neural network.

Neural Networks

The models predict quality scores using as labels the GMM loglikelihood for the between-subject case. The inputs to the networks are the skeletal joints data with 117 dimensions.

Prepare_Data_for_NN: reads the full-body skeletons data and performs initial pre-processing, such as alignment and centering.

Prepare_Labels_for_NN: calculates movement quality scores based on GMM encoded sequence of reduced dimensionality by autoencoder neural network.

SpatioTemporalNN_Vicon: is the proposed model for predicting movement quality scores.

CNN_Vicon: is a basic model for predicting quality scores consisting of convolutional layers (in the paper referred to as Deep CNN model).

RNN_Vicon: is a basic model for predicting quality scores consisting of recurrent LSTM layers (in the paper referred to as Deep LSTM model).

Autoencoder_Dims_Reduction: is used for reducing the dimensionality of the data from 117-dimensions to 4 dimensions.

SpatioTemporalNN_Kinect: is an implementation of the proposed deep learning model for predicting quality score on skeletal data collected with the Kinect v2 sensor from the open dataset KIMORE. The implementation details are provided in the paper.

Distance Functions

Euclidean_btw_subjects: calculates Euclidean distance on the full-body skeleton data for the between-subject case.

Euclidean_wthn_subjects: calculates Euclidean distance on the full-body skeleton data for the within-subject case.

Mahalanobis_btw_subjects: calculates Mahalanobis distance on the full-body skeleton data for the between-subject case.

Mahalanobis_wthn_subjects: calculates Mahalanobis distance on the full-body skeleton data for the within-subject case.

DTW_btw_subjects: calculates DTW distance on the full-body skeleton data for the between-subject case.

DTW_wthn_subjects: calculates DTW distance on the full-body skeleton data for the within-subject case.

Var_Euclidean_btw_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimensional data for the between-subject case.

Var_Euclidean_wthn_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimensional data for the within-subject case.

Var_Mahalanobis_btw_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimensional data for the between-subject case.

Var_Mahalanobis_wthn_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimensional data for the within-subject case.

Var_DTW_btw_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case.

Var_DTW_wthn_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case.

Var_loglikelihood_btw_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimensional data for the between-subject case.

Var_loglikelihood_wthn_subjects: uses maximum variance to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimensional data for the within-subject case.

PCA_Euclidean_btw_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimensional data for the between-subject case.

PCA_Euclidean_wthn_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimensional data for the within-subject case.

PCA_Mahalanobis_btw_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimensional data for the between-subject case.

PCA_Mahalanobis_wthn_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimensional data for the within-subject case.

PCA_DTW_btw_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case.

PCA_DTW_wthn_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case.

PCA_loglikelihood_btw_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimensional data for the between-subject case.

PCA_loglikelihood_wthn_subjects: uses PCA to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimensional data for the within-subject case.

Encoder_Euclidean_btw_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimension data for the between-subject case.

Encoder_Euclidean_wthn_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates Euclidean distance on the low-dimension data for the within-subject case.

Encoder_Mahalanobis_btw_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimension data for the between-subject case.

Encoder_Mahalanobis_wthn_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates Mahalanobis distance on the low-dimension data for the within-subject case.

Encoder_DTW_btw_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case.

Encoder_DTW_wthn_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case.

Encoder_Loglikelihood_btw_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimension data for the between-subject case.

Encoder_Loglikelihood_wthn_subjects: uses an autoencoder neural network to reduce the dimensionality of the data, and afterward calculates GMM loglikelihood on the low-dimension data for the within-subject case.

Utility Functions

EM_boundingCov: trains the parameters of a GMM using a recursive Expectation-Maximization (EM) algorithm. After each EM step, the covariance matrices are bounded to avoid numerical instability.

EM_init_regularTiming: initializes the parameters of a Gaussian Mixture Model (GMM) by using *k*-means clustering algorithm.

gausPDF: computes the Probability Density Function (PDF) of a multivariate Gaussian represented by means and covariance matrix.

loglik: computes the loglikelihood of a GMM model.

PlotGMM1: plots the mean and covariance matrix of a GMM model.

Results

The folder contains several figures produced by the above-described functions that were more important for this study.

DTW_Distance_BS_MV: produced by Var_DTW_btw_subjects shows the DTW distance for the data with reduced dimensionality by using the maximum variance approach for the between-subjects case.

DTW_Distance_BS_NDR: produced by DTW_btw_subjects shows the DTW distance for the full-body skeletal data for the between-subjects case.

Euclidean_Distance_BS_NDR: produced by Euclidean_btw_subjects shows the Euclidean distance for the full-body skeletal data for the between-subjects case.

GMM_Loglikelihood_BS_ENC: produced by Encoder_Loglikelihood_btw_subjects shows the GMM loglikelihood for the data with reduced dimensionality by using the autoencoder model for the between-subjects case.

Mahalonobis_Distance_BS_PCA: produced by PCA_Mahalanobis_btw_subjects shows the Mahalanobis distance for the data with reduced dimensionality by using PCS for the between-subjects case.

GMM_Encoded_Movements: produced by Prepare_Labels_for_NN shows the GMM encoded correct sequences of autoencoder reduced data.

GMM_Loglikelihood_Scores: produced by Prepare_Labels_for_NN shows the GMM loglikelihood for the correct and incorrect sequences of autoencoder reduced data.

GMM_Movement_Quality_Scores: produced by Prepare_Labels_for_NN shows the scaled GMM loglikelihood for the correct and incorrect sequences of autoencoder reduced data.

CNN_Vicon_Scores: produced by CNN_Vicon shows the predicted movement quality scores by the basic CNN model for the correct and incorrect sequences.

RNN_Vicon_Scores: produced by RNN_Vicon shows the predicted movement quality scores by the basic RNN model for the correct and incorrect sequences.

SpatioTemporalNN_Vicon_Scores: produced by SpatioTemporalNN_Vicon shows the predicted movement quality scores by the proposed spatio-temporal model for the correct and incorrect sequences.

SpatioTemporalNN_Kinect_Scores: produced by SpatioTemporalNN_Kinect shows the predicted movement quality scores by the proposed spatio-temporal model for the correct and incorrect sequences captured with the Kinect v2 sensor.