Averaged Hidden Markov Models in Kinect-Based Rehabilitation System

Aleksandra Postawka(✉) and Przemysław Śliwiński

Faculty of Electronics, Wroclaw University of Science and Technology, Wrocław, Poland
{aleksandra.postawka,przemyslaw.sliwinski}@pwr.edu.pl

Abstract. In this paper the Averaged Hidden Markov Models (AHMMs) are examined for the upper limb rehabilitation purposes. For the data acquisition the Microsoft Kinect 2.0 sensor is used. The system is intended for low-functioning autistic children whose rehabilitation is often based on sequences of images presenting the subsequent gestures. The number of such training sets is limited and the preparation of a new one is not available for everyone, whereas each child requires the individual therapy. The advantage of the presented system is that new activities models could be easily added.

The conducted experiments provide satisfactory results, especially in the case of single hand rehabilitation and both hands rehabilitation based on asymmetric gestures.

Keywords: Autistic children · Rehabilitation
Hidden Markov Models · Averaged Hidden Markov Models
Microsoft Kinect 2.0 · Depth sensor

1 Introduction

The necessity of rehabilitation is usually considered to be useful only after accidents or injuries. However, it is also needed in some cognitive impairments, such as autism. The rehabilitation of low-functioning autistic children is often based on sequences of images presenting the subsequent gestures to be performed. The number of such rehabilitation sets is limited and the preparation of a new one is not available for everyone. It is a well known fact that every autistic person is different and needs the individual approach [1–3]. Hence, the individuals with autism may need significantly different rehabilitation exercises.

In the literature there are numerous examples of rehabilitation systems. The one of mostly considered issues is a post-stroke rehabilitation as that strokes are the often a cause of motor deficits. Regenbrecht *et al.* proposed the system ART for the treatment of upper limb dysfunctions [4]. The system uses the augmented reality and computer games based approach in order to increase patients motivation. The post-stroke rehabilitation system has been also developed by Kuttuva *et al.* [5]. In this tool called Rutgers Arm the virtual reality (VR) technology

L. Rutkowski et al. (Eds.): ICAISC 2018, LNAI 10842, pp. 229–239, 2018.
https://doi.org/10.1007/978-3-319-91262-2_21

is used. The another example is a game-based system for upper limbs rehabilitation presented by Pastor *et al.* [6]. For post-stroke patients rehabilitation the wearable wireless sensors have been used as well [7].

Along with the development of the depth sensors technology the Microsoft Kinect became an increasingly used device in rehabilitation. Clark *et al.* have confirmed Kinect to be successfully used for assessing the postural control [8]. Scherer *et al.* developed the Kinect-based system for injured athletes [9]. Another example for using Kinect in rehabilitation is a system Kinere, which has been applied for patients suffering from (1) severe cerebral palsy and (2) acquired muscle atrophy [10]. Kusaka et al. used Kinect to develop a rehabilitation system for patients with hemiplegia [11]. The Kinect-based rehabilitation system for home usage was designed by Su *et al.* [12]. Kinect was also used in the system intended for patients with body scheme dysfunctions and left-right confusion [13].

Despite the great number of research dealing with rehabilitation, we believe that there is still a lack of the system which would make it possible to easily add a new rehabilitation task. Such a feature would be very useful in the very individualized in their nature therapies for autistic persons.

The other motivation for this research is that Hidden Markov Models (HMMs) have been never used, to the best of authors knowledge, in the task of rehabilitation, in spite of the fact that the left-to-right HMM [14] models, which preserve the information about the observation symbols order, seem to be a valuable tool for the purpose of motion tracking and evaluation. The Averaged HMMs [15], composed of multiple left-to-right HMMs, combine the features of all component models, thus the most commonly occurring features could be retrieved from such a final model. Moreover, the observation symbol distribution in states is a property that describes movements' noise and uncertainty in a very natural way.

- First of all the symbol distribution in AHMM contains only motions that really occurred in the therapists movements.
- Secondly, the multitude of learning sequences ensures that a wide variety of proper movements is included.

In this paper the usefulness of Averaged Hidden Markov Models (AHMMs) in the rehabilitation exercises have been examined. The 13 activity AHMM models have been used for the upper limbs rehabilitation. For the data acquisition the Microsoft Kinect 2.0 depth sensor is used.

The paper is organized as follows. The notation is introduced in Sect. 2. The Sect. 3 contains the description of methods used in the research. The application and usage examples have been presented in Sect. 4. The Sect. 5 contains the overall conclusions and plans for the future.

2 Notation

In the paper the following notation is used for HMMs:

$\lambda = \{A, B, \pi\}$ - the complete parameter set for HMM,
N - the number of states,
M - the number of observation symbols,
T - the length of observation sequence,
$A = \{a_{ij} : i, j \in \{1, \dots, N\}\}$ - the state transition matrix,
$B = \{b_{ij} : i \in \{1, \dots, N\}, j \in \{1, \dots, M\}\}$ - the probability distribution matrix for observed symbols,
$\pi = \{\pi_i : i \in \{1, \dots, N\}\}$ - the initial state distribution vector,
$O = O_1, O_2, \dots, O_T$ - the observation sequence,
$O_{j:k}$ - the part of observation sequence including symbols from j-th to k-th, inclusive.

For AHMMs the following notation is used:

D - the number of component models,
$x^{(d)} : d \in \{1, \dots, D\}$ - the value x from d-th component model.

3 Methods

The rehabilitation module is based on Averaged Hidden Markov Models (AHMMs) [15], briefly described below. In order to decide whether the new motion is correct or not, the rehabilitation models extend the methods for action recognition in HMM. These algorithms are listed in Sect. 3.2. The other methods used for rehabilitation are described in Sects. 3.3 and 3.4.

3.1 Averaged Hidden Markov Models

Each activity model is created from multiple learning sequences. In the later stages of AHMM generation each single learning sequence becomes the base for a one component HMM model. Thus the number D of learning sequences is equal to the number of component models.

One of the learning sequences is chosen as a *pattern sequence* and defines all models structure, i.a. the number of states. The pattern sequence is used for the *base model* definition. The base model is a left-to-right HMM with states matched to subsequent observation symbols, thus in this stage the model could be reduced to the simple Markov chain.

Each of the rest of component models is computed based on the base model and the corresponding learning sequence. Such a *child model* has a structure *similar* to the base model:

- the child model is also a left-to-right HMM model,
- the base and child models have the same number of states,
- the same observation symbols in base and child models occur in the same states, taking under consideration the symbol order.

The detailed algorithm for base and child models parameters computation is described in the previous work [15]. At this stage there is D similar left-to-right HMMs.

Finally, all the component models are simply averaged using the Eqs. (1), (2) and (3). In consequence, we obtain one resultant Averaged HMM model which generates each of its learning sequences $O^{(i)}$ with the probability $P(O^{(i)}|\lambda) > 0$.

$$\bar{a}_{ij} = \sum_{d=1}^{D} \tfrac{1}{D} \cdot a_{ij}^{(d)} \tag{1}$$

$$\bar{b}_{ij} = \sum_{d=1}^{D} \tfrac{1}{D} \cdot b_{ij}^{(d)} \tag{2}$$

$$\bar{\pi}_{i} = \sum_{d=1}^{D} \tfrac{1}{D} \cdot \pi_{i}^{(d)} \tag{3}$$

3.2 Action Recognition

The task of rehabilitation could be considered as the subproblem of the real-time recognition problem. In both cases the information whether the activity is completed or not is crucial. The posterior probability $P(O|\lambda)$ is not a sufficient indicator, while each beginning part of the recognized sequence is also recognized, i.e.

$$\forall_{1 \le t < T} P(O_{1:T}|\lambda) > 0 \Longrightarrow P(O_{1:t}|\lambda) > 0.$$

Therefore the method developed for real-time recognition [16] has been used. The N_R value (the real number of states, which is different for each of the component models) has been added to the model during the learning phase as the id of the last state that is always accessed while recognizing any of learning sequences. It means that the sequence ending in the state with lower id than N_R is not complete. The last state is estimated by the Viterbi algorithm [14] based on the observation symbol sequence.

Because of short activities, which consist of less than 4 symbol changes and where noise might change the recognized class (activity id), the additional condition has been added. The last symbol O_T in the considered sequence has to be probable to occur in the last state in the model, i.e. $b_{NO_T} > 0$.

The complete sequence recognized by model λ fulfills all the three conditions for this model.

3.3 Rehabilitation

The idea of rehabilitation using HMMs is based on displaying the most probable symbol in the next most probable state based on the actual state. The actual state is estimated using the Viterbi algorithm. In general the rehabilitation problem could be stated as a set of following equations. The first state is chosen as the most probable one due to the initial state distribution using the Eq. (4). Each next state is chosen based on the probability transition matrix and the actual state (Eq. (5)). While having the most probable state, the next symbol to be displayed is calculated by Eq. (6).

$$\text{Find state } i \in \{1, \dots, N\} : \forall_{j \in \{1,\dots,N\}} \pi_i \geq \pi_j \tag{4}$$

$$\text{For the state } j \text{ find state } i \in \{1, \dots, N\} : i \neq j \wedge \forall_{k \in \{1,\dots,N\}} a_{ji} \geq a_{jk} \tag{5}$$

$$\text{For the state } i \text{ find symbol } k \in \{1, \dots, M\} : \forall_{l \in \{1,\dots,M\}} b_{ik} \geq b_{il} \tag{6}$$

Because of the fact that in evaluation problem [14] the posterior probability tends to zero exponentially along with the increase of the number of observation symbols, the logarithmized value is calculated instead. However, during the long rehabilitation process (e.g. the motions are very slow) the range of double is also exceeded and $log(P(O|\lambda)) = -\infty$. Therefore, in the system only the symbol changes are registered. Sometimes the list of symbols to be displayed calculated by Eqs. (4–6) may consist of series of the same symbol - the most probable symbol in the next most probable state may be the same symbol as the most probable symbol in the previously most probable state. While we register only the symbol changes, such situation detection had to be introduced. In order not to reduce the intelligibility of the algorithm this case has been marked in red in the Fig. 1. The variable *skippedSameSymbol* is set to the number of next equal symbols. If this variable is greater than zero then exceptionally the adjacent equal symbols (in a number equal to this variable) are added to the symbol history. In further discussion this case will be omitted.

The diagram showing the complete rehabilitation algorithm is presented in Fig. 1. First of all the HMM symbol is calculated depending on the chosen motion (right–, left– or both–handed model) according to the algorithm described in [17]. The information about the previous symbol (performed motion) and previously displayed symbol is saved from the previous iteration. Also the zero-one information whether the motion was correct or not (due to the value $log(P(O|\lambda))$) is remembered.

Secondly, if the actual symbol is the same as in the previous iteration, then the old values of (1) the next symbol to be displayed and (2) correctness of motion are returned. Otherwise, the symbol is added to the history and remembered as the previous symbol for the next iteration. The model is evaluated using the forward algorithm [14] based on the actual observation symbol history.

Next, if $log(P(O|\lambda)) \neq -\infty$ then the actual motion is marked as correct and the state sequence is estimated using the Viterbi algorithm. The next state is estimated based on the last decoded state using the Eq. (5). Next symbol to be displayed is calculated based on the estimated future state using the Eq. (6). Otherwise, if $log(P(O|\lambda)) = -\infty$ then the actual motion is marked as incorrect, the symbol is rolled back from the history and the algorithm returns the same values as in the previous iteration.

3.4 Hand Coordinates Estimation

The only information about the next movement obtained from the HMM is the observation symbol, i.e. the natural number in the range of $\{1; M\}$. For the rehabilitation module the visualization is the one of the most important features, thus the hands coordinates have to be calculated. In order to estimate these

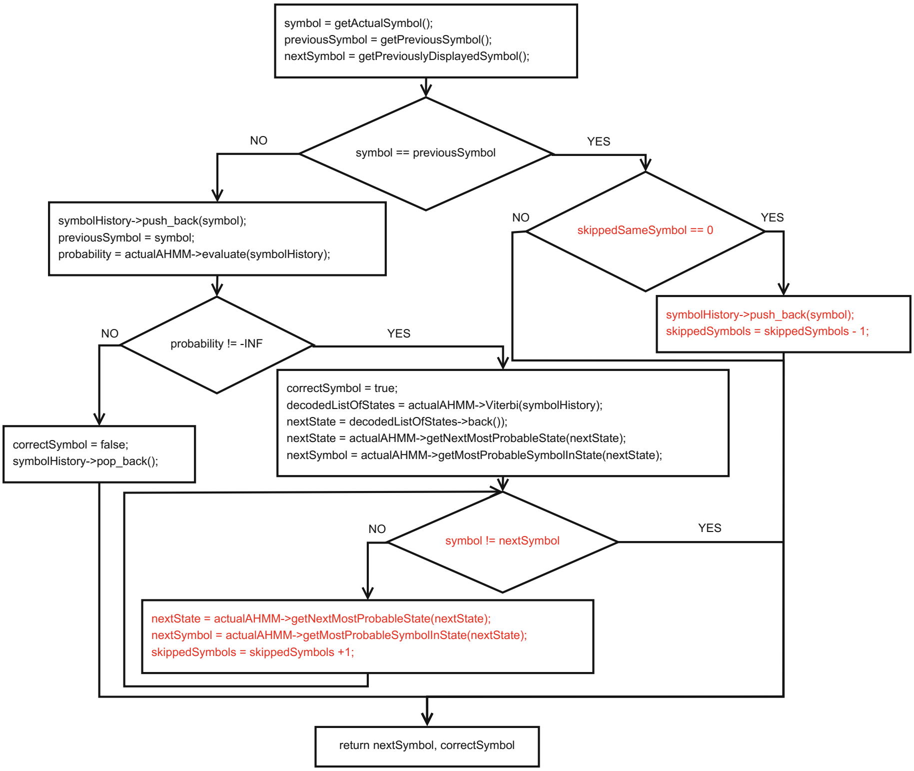

Fig. 1. Rehabilitation flowchart

parameters the inverse operations than in the hand position classifier described in [17] (Eqs. (1) and (2)) are needed.

Firstly, for each feature h_i and the HMM hand symbol s_N the id of the including interval $z_i \in \{0; m_i - 1\}$ (m_i is the number of intervals that the range of values for h_i had been divided into) is calculated using the Eq. (7). The r_i is an auxiliary variable.

$$\begin{cases} r_0 = s_N \\ z_i = r_{i-1} div \prod_{k=i+1}^{N} m_k \\ r_i = r_{i-1} mod \prod_{k=i+1}^{N} m_k \end{cases} \tag{7}$$

Secondly, based on the minimum $h_{i_{MIN}}$ and maximum $h_{i_{MAX}}$ value in the range, the value of h_i is estimated (Eq. 8). In order to minimize the error for the singular feature value estimation the middle of the range is chosen.

$$h_i = h_{i_{MIN}} + (z_i + 0,5) \cdot \frac{h_{i_{MAX}} - h_{i_{MIN}}}{m_i} \tag{8}$$

Finally, based on the estimated feature values and the length of the arm, the hand coordinates are calculated. The features used in the classification were as follows:

- h_1 - an angle between projection of the vector $\boldsymbol{v}$ to the OXZ plane and the x axis,
- h_2 - an angle between the vector $\boldsymbol{v}$ and the y axis,
- h_3 - a relative length of the vector $\boldsymbol{v}$ (the quotient of $|\boldsymbol{v}|$ and the length of the whole arm),

where $\boldsymbol{v}$ is the radius vector connecting the shoulder and the hand.
Therefore, in order to calculate the hand's coordinates, the point $(|\boldsymbol{v}|, 0, 0)$ is rotated by the calculated angles in the OXY and OXZ planes.

4 Application

The rehabilitation module is a part of the more complex system designed for children with autism [17]. The application is designed to track one- or both-hands movements. The list of AHMM activities models chosen for rehabilitation is presented in Table 1. The list has been divided into left-, right- and both-handed activities. The system is fully scalable, while the files with models could be easily added or deleted from the models directory.

Table 1. The list of modeled activities used for rehabilitation

Left-handed activities	Right-handed activities	Both-handed activities
Left arm twisting forward	Right arm twisting forward	Both hands twisting forward
Left arm twisting backward	Right arm twisting backward	Both hands twisting backward
Raising and lowering left hand	Raising and lowering right hand	Raising and lowering both hands
		Clapping hands
		Clapping hands over the head
		Crawl forward
		Crawl backward

In the rehabilitation mode the two skeletons are displayed - the pattern to follow and the actual motion. The skeletons coordinates are normalized as in [17], so that the *Spine Shoulder* joint overlaps in both cases. The complete body coordinates set is retained in the file, but the hand coordinates are calculated based on the next symbol to be displayed (algorithm described in Sect. 3.4). The elbow coordinates for the symbols are tabulated. The actual hand position is additionally surrounded by a circle which changes the color depending on whether the motion is correct (green) or not (red) according to the model. Since in the 2D picture it is difficult to guess the limb distance from the camera, the displayed color changes. If the joint is further from the camera than the *Spine Shoulder* joint (greater z value) then the joint connection is painted in blue.

After choosing the rehabilitation mode and the activity AHMM model, the first motion is displayed. Depending on the patient's movements the next symbols (hands positions) are estimated and displayed. An example for the rehabilitation is presented in Fig. 2. In the picture the most important fragments for the recording of twisting left arm forward have been included. An example of the incorrect motion is presented in Fig. 3(a) - the hand is surrounded with the red circle.

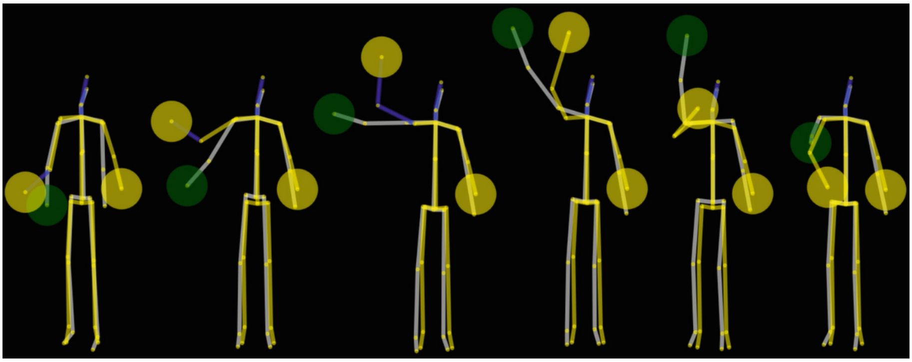

Fig. 2. Rehabilitation with the exercise: twisting left arm forward (Color figure online)

In the case of single hand rehabilitation, the future movements estimated by the AHMM model coincide with expectances. The speed of motions does not affect the result of action recognition, however it has the influence on the motion evaluation $P(O|\lambda)$.

The issue of both hands rehabilitation is much more complicated. The data taken from the real world abound in noises, for example it is nearly impossible to perform the strictly symmetric motion by each of the hands. Therefore, in such a symmetric movements like twisting arms forward, the most probable future symbol chosen by the algorithm often corresponds to asymmetric hands position. In such a case even if in the real motion (learning sequences) the hands positions did not differ much, the algorithm (Sect. 3.4) estimates the coordinates strongly asymmetric. The example of asymetric hands position prediction is presented in Fig. 3(b), while the Fig. 3(c) presents the next stage of the same activity which is symmetric. The problem is not visible if there is no symmetry in both hands movement. The problems with visualization however do not affect the motion assessment.

Summarizing, the applied algorithms meet the requirements, especially when it comes to the one–handed motions. The bottleneck of this rehabilitation system is the lack of legible visualization, as that the position of the 3D point presented on the 2D screen is ambiguous.

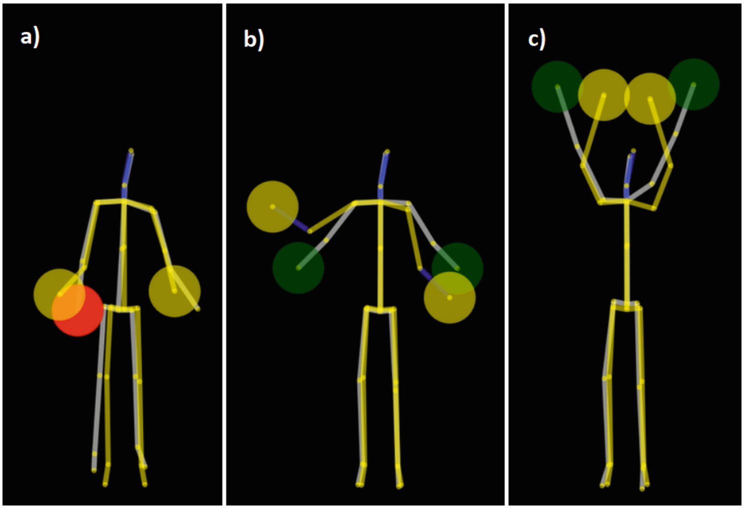

Fig. 3. (a) The one–hand incorrect motion (red circle) (b) The both–hand asymmetric movement (c) The both–hand symmetric movement (Color figure online)

5 Conclusions and Future Work

In this paper the Averaged Hidden Markov Models were examined for the rehabilitation purposes. The one– and two–handed activity models were taken under the consideration.

The conducted experiments indicate that AHMMs provide the satisfactory results for the rehabilitation purpose as the motion is tracked and assessed properly. In the case of single hands rehabilitation and both hands rehabilitation based on motions without symmetry, the next movement prediction coincides with the expectations.

The visualization of the future movement sometimes could be unintuitive, as that the predicted next hands positions for the symmetric both hands motion could have no symmetry. Also the position of the 3D point presented on the 2D screen is ambiguous. The symbolic representation (skeleton) also seems to be too abstract for children with autism. For the practical usage of the examined algorithms the visualization methods need to be improved. On the other hand, such a symbolic representation respects the privacy as that only the skeleton joints are taken under consideration.

The advantage of the system is that the new activities could be easily added. This function could be especially important in the case of autistic children rehabilitation, as that each child with autism needs individual therapy.

Acknowledgment. This work was supported by the statutory funds of the Faculty of Electronics 0401/0159/17, Wroclaw University of Science and Technology, Wroclaw, Poland.

References

1. Seach, D., Lloyd, M., Preston, M.: Supporting Children with Autism in Mainstreem Schools. The Questions Publishing Company Ltd., Birmingham (2003). ISBN 83-60215-17-0
2. Barry, A.: Some people think that every person with autism is like Rain Man, or a wizard at maths. Thejournal (2017). http://www.thejournal.ie/autism-aspergers-ireland-3297234-Mar2017/
3. Autism Awareness - Frequently Asked Questions About Autism. Staffordshire Adults Autistic Society. http://www.saas.uk.com/p/autism-awareness-questions.php
4. Regenbrecht, H., Hoermann, S., McGregor, G., Dixon, B., Franz, E., Ott, C., Hale, L., Schubert, T., Hoermann, J.: Visual manipulations for motor rehabilitation. Comput. Graph. (Pergamon) **36**(7), 819–834 (2012)
5. Kuttuva, M., Boian, R., Merians, A., Burdea, G., Bouzit, M., Lewis, J., Fensterheim, D.: The rutgers arm, a rehabilitation system in virtual reality: a pilot study. CyberPsychol. Behav. **9**(2), 148–152 (2006)
6. Pastor, I., Hayes, H.A., Bamberg, S.J.M.: A feasibility study of an upper limb rehabilitation system using Kinect and computer games. In: 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 1286–1289 (2012)
7. Chee, K.L., Chen, I.M., Zhiqiang, L., Yeo, S.H.: A low cost wearable wireless sensing system for upper limb home rehabilitation. In: 2010 IEEE Conference on Robotics, Automation and Mechatronics, pp. 1–8 (2010)
8. Clark, R.A., Pua, Y.H., Fortin, K., Ritchie, C., Webster, K.E., Denehy, L., Bryant, A.L.: Validity of the Microsoft Kinect for assessment of postural control. Gait Posture **36**(3), 372–377 (2012)
9. Scherer, M., Unterbrunner, A., Riess, B., Kafka, P.: Development of a system for supervised training at home with Kinect V2. Procedia Eng. **147**, 466–471 (2016)
10. 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)
11. Kusaka, J., Obo, T., Botzheim, J., Kubota, N.: Joint angle estimation system for rehabilitation evaluation support. In: 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1456–1462 (2014)
12. Su, Ch.J., Chiang, Ch.Y., Huang, J.Y.: Kinect-enabled home-based rehabilitation system using dynamic time warping and fuzzy logic. Appl. Soft Comput. **22**, 652–666 (2014). Elsevier B.V
13. González-Ortega, D., Díaz-Pernas, F.J., Martínez-Zarzuela, M., Antón-Rodríguez, M.: A Kinect-based system for cognitive rehabilitation exercises monitoring. Comput. Methods Programs Biomed. **113**, 620–631 (2014)
14. Rabiner, L., Juang, B.: An introduction to hidden Markov models. IEEE ASSP Mag. **3**, 4–16 (1986)
15. Postawka, A.: Exercise recognition using averaged hidden Markov models. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2017 Part II. LNCS (LNAI), vol. 10246, pp. 137–147. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59060-8_14

16. Postawka, A.: Real-time monitoring system for potentially dangerous activities detection. In: Proceedings of the 22nd International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 1005–1008. IEEE Xplore Digital Library (2017)
17. Postawka, A., Śliwiński, P.: A Kinect-based support system for children with autism spectrum disorder. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016 Part II. LNCS (LNAI), vol. 9693, pp. 189–199. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39384-1_17