Action Pose Recognition from 3D Camera Data Using Inter-frame and Inter-joint Dependencies

Blake Capella*, Deepak Subramanian†, Roberta Klatzky‡, Daniel Siewiorek§
*Department of Electrical and Computer Engineering
The College of New Jersey, Ewing, New Jersey
capellb1@tcnj.edu
†Department of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, Michigan
deepaksm@umich.edu
‡Department of Psychology
§Department of Electrical and Computer Engineering
Carnegie Mellon University, Pittsburgh, Pennsylvania
‡klatzky@cmu.edu
§dps@cs.cmu.edu

***Abstract*—We developed a classifier for back stretching body poses used in therapy. Joint coordinates were collected and normalized to adjust for different body proportions and speeds. Initial classification on the entire normalized data set established a baseline of 25% accuracy, relative to 17% for chance. A second-stage classifier was developed to use inter-frame movement kinematics to isolate the core frames where the final static pose was achieved. Combined with higher-order features from inter-joint dependencies, the modifications produced an accuracy of 63%.**

***Index Terms*—Machine Vision,Activity Recognition, Machine Learning, Neural Networks, Signal Processing Algorithms**

I. Introduction

Human action recognition (HAR) is a challenge for today's computing systems, especially in comparison to the capability of the human brain. Given a video or frame of a single human action, computers are challenged to perform a classification that is rapidly accomplished by human perceptual systems. HAR requires a flexible system capable of adapting to individual stature and movement variations, dynamic action definitions, and many other variables. Although HAR poses many challenges, its applications in computer human interaction, surveillance, and many other fields make it an intriguing research goal.

With the commercialization of depth camera technologies (e.g., Microsoft Kinect) the health industry has access to motion analysis and remote viewing devices. This new technology, when combined with HAR, has potential to reduce therapist workloads and improve patient motivation [1], [2]. In addition, disease or condition-specific applications have shown promise for event and symptom detection [3]. Despite the success of these approaches, the Kinect has shortcomings. Kaewplee et al.'s study of Muay Thai [4] showed that in full body movements the Kinect's limited accuracy results in jitter, dead limbs, and joint swapping. In this study we propose a new approach to improve the accuracy of different Kinect models with potential applications in other HAR systems.

Within the field of HAR, there exist different approaches. Zhu. et al. [5] identifies segmented human action recognition (SHAR) and continuous human action recognition (CHAR). SHAR approaches work on datasets composed of single subject and single action, where the primary objective is to classify the action. Examples of common SHAR approaches are codebook [6], [7], histogram [8]–[10], and relative joint location [11], [12]. In this study, we focus on SHAR in total body exercises for use in rehabilitation and therapy.

In CHAR, researchers focus on detecting when an action is taking place rather than which specific action is occurring. The most prominent CHAR approach involves the use of sliding windows to select action periods. Multiple variations of this approach exist, each proposing unique ways of identifying window start and end points [13]–[15].

Drawing from this research, our approach combines continuous and segmented HAR algorithms. Using a sliding window approach common in CHAR, we isolate the frames of the final static pose contained in a single subject, single exercise dataset. Due to the lack of appropriate datasets, a portion of the study is devoted to the creation of a full body exercise dataset. Similar to [15], our window is based on periods of activity. Here, we use velocity data from near-contemporaneous frames to detect static hold periods. After the application of the sliding window approach, we use a more conventional SHAR algorithm to classify each action.

In the next part of the paper, we describe our approach to achieving classification of the exercise sequences. Data were collected from the Kinect as joint coordinates and then normalized to adjust for variations in individuals' stature and movement. An initial neural-net model was then used to select hyperparameters appropriate to the data set; this also

Contributions of the first two authors are equivalent. We acknowledge support from the National Science Foundation (NSF) under grant number CNS-1518865. We would like to thank Dr. Asim Smailagic for his support throughout the project.

established the baseline level of performance possible when all frames from the data set were randomly intermixed. Finally, movement kinematics were used to isolate the core frames from the data set in which the exercise was achieved. Higher-order features extracted from those frames were used as a basis for a final classifier, using the hyperparameters identified by the baseline model.

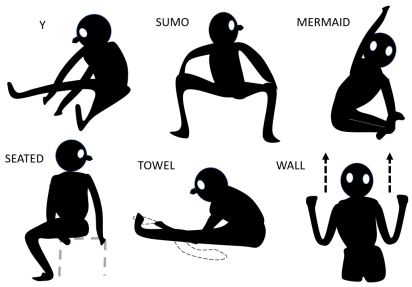

Fig. 1. Stretches included in Dataset

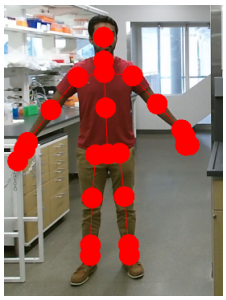

Fig. 2. Sites Recorded in Data Collection; not visible are individual thumb joints

II. Methods

A. Data Collection

Pre-existing libraries, specifically the KinectPV2 library, were used to build an efficient platform for data collection and processing [16]. The six full body stretching exercises shown in Figure 1 were used to create a dataset composed of exercise frame sequences. Each of these stretches consists of a movement period, a hold in place, then a return to a neutral position. The stretches all incorporated placement of the legs and feet, followed by movement of the upper body intended to stretch the back, but in ways that distinguished each exercise [17]. For example, only *Sumo* has abducted legs with bent knee and a torsional twist around the waist. Nine young adults (age 18-21, four females and five males) volunteered, without remuneration, to perform the stretches to establish the data set. Each stretch began with a short instructional video, after which the experimenter signaled the participant to perform 10 repetitions of that stretch, returning to a neutral pose between each. Upon completion of the repetitions for one stretch, the video for the next was shown. Participants were permitted to ask clarifying questions at any time. The Microsoft Kinect v2 camera collected the 3D position of 25 of the subject's joints or fixed body locations, shown in Figure 2, at approximately 30 frames per sec. Further details on the data set can be found in Table 1. The data set is mounted at XXXXXXXXXXXX

TABLE I
Dataset Size

Measure	Number
Total Frames Collected	119,999
Total Joint Positions Collected	2.99×10^6
Average Frames per Exercise Repetition	221.81

B. Data Normalization

Participants vary considerably in stature and build and move at different speeds. Thus there is the need to adjust for individual variations. The raw three-dimensional position data were normalized by three calculations described in Table 2. The position coordinates, originally in a reference frame defined by the camera, were re-calculated relative to a framework centered on the subject's chest, to allow for comparisons of individuals of different stature. Then velocity for each coordinate was computed to use as a variable in the classifier, under the assumption that it would be critical to differentiating the exercise set. Both variables were converted to within-participant z-scores, allowing comparison across individuals with different movement rates and distances.

III. Baseline Neural Network Parameterization

An initial neural-net model was constructed in order to select model parameters and establish a baseline level of classification on the entire data set of normalized coordinates. Candidate parameterizations were implemented in Python using Tensorflow, Google's machine learning library. A model was constructed by fixing a set of hyperparameters constituting values of the following model training variables: network architecture (4), activation function (4 candidates), learning rate (4), and training epochs (3). The specific values tested were selected to survey the parameter space. To select final hyperparameter values, model training runs were used to evaluate performance.

TABLE II
Normalization Calculations

Name	Transformation	Description
Position Coordinate Normalization	$X_{Normal} = X - X_{Chest}$ $Y_{Normal} = Y - Y_{Chest}$ $Z_{Normal} = Z - Z_{Chest}$	Linearly scaled the Kinect's coordinate system to reflect an origin at the subject's chest
Velocity Calculation	$v[n] = \frac{x[n]-x[n-5]}{5}$	Computed velocity as the change in position across 5 frames
Velocity and Position Standardized (z) Score (computed for each joint of each individual within each exercise)	$z = \frac{x-\mu}{\sigma}$	Converted the position and velocity data to z-scores, pooling the observations across all frames of a single exercise (10 repetitions). This standardization is computed separately for each combination of participant, joint and exercise.

For purposes of development, a sample of 60 fully executed exercises (repetitions) was constructed by randomly sampling 10 repetitions of each exercise type. Because a repetition is the collection of frames that were captured for one fully executed exercise, the number of frames in each repetition is variable. No other specifications of the data besides exercise type were considered when taking the subsample (e.g., number of frames, noise levels, subject, time of data collection, etc.). Once the subsample was constructed, the 60 repetitions were further divided into two categories, used for training and testing candidate neural nets that would classify exercises from single frames. Independently for each test of a candidate model, the procedure was as follows: (i) Initially, 70%, or 42 full exercise repetitions from the subsample, were randomly chosen for use as training data. The remaining 30%, or full 18 exercise repetitions, constituted the testing data and were withheld from the model until completion of training. (ii) Within the training repetitions, the individual frames were ungrouped and randomized as to order. The input associated with each frame comprised the raw coordinates of each of the 25 body parts, constituting 75 data points per frame. (iii) The model was trained 10 times and its average accuracy computed.

The hyperparameters that yielded the best accuracy across the 10 training runs for the given model were an architecture of 4 fully connected layers of 30 nodes with an output layer of Softmax and an Adam optimization algorithm, ReLU activation, a learning rate of .001, and 1000 training epochs.

Run with a confidence criterion of .20, the accuracy from the model parameterized as above reached only 25%. It should be noted, however, that the evaluation uses a data stream that mixes frames from all the exercises in random order. This test ignores the fact that the data source is a sequence of frames in a single exercise and hence does not capitalize on inter-frame dependencies. In Section 4, we used the inter-dependence of frames within an exercise to refine the problem and improve classification accuracy.

IV. Isolating Static Holds from Inter-frame Kinematics

A single stretch exercise consists of moving into a static pose, holding the pose, and releasing. The distinctive features of the exercise are mainly to be found in the static hold. To isolate the period of the static hold, it is necessary to identify and remove the periods of transition in which the stretch was entered and exited. Removing transitions should improve the model not only by isolating the defining features of the stretch, but because the Kinect v2 camera, as with motion tracking systems in general, has the least precision when the observed person is in motion [4]. Thus eliminating periods of movement also removes the highest error rates in position measurement, and ultimately, reduces the noise in the data entered for classification by the model.

To determine an algorithm for isolating transitions, we examined the standardized velocity data of each joint at each frame in a particular exercise, pooling data across the 10 repetitions for all participants. This calculation also pooled the

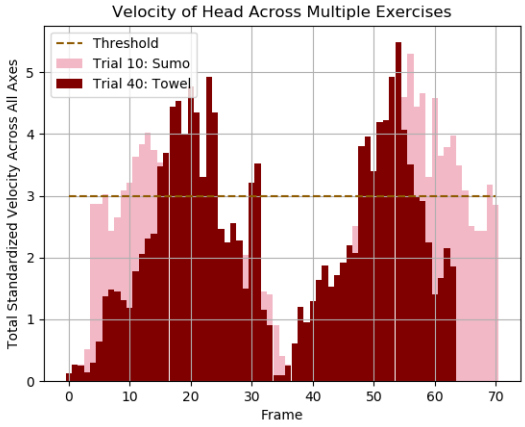

Fig. 3. Plot of the Total Standardized Velocity Across all Axes for *Mermaid* and *Sumo* Exercises at the Head

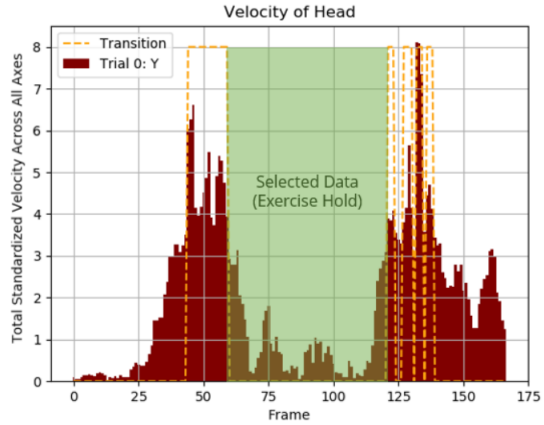

Fig. 4. Selection Algorithm Applied to the Velocity Histogram for Y

standardized velocity data across each axis, creating a metric representing the absolute magnitude of 3D movement from the participant's joint. The new data from this phase was used in all model training and evaluation phases in place of the unfiltered data. Note that standardization controls for different movement speeds of participants. Figure 3 shows the total standardized velocity across all axes for the head/neck joint, by frame within the movement, for the *Sumo* and *Mermaid* stretches. It can clearly be seen that periods of high velocity, presumably transitions, occur between static pose periods with lower velocity. This suggests that thresholding the data by velocity could be used to isolate the transitions. Accordingly, the histogram of the standardized velocity data of a single joint at each frame in a particular exercise was computed, pooling data across axes, as seen in Figure 3. Examination of the data suggested that a threshold where the total standardized velocity across all axes must exceed a combined score of 3.0 would identify the high velocities corresponding to movement of the joint as a transition occurred between stretch poses. The selected threshold used by the algorithm is also shown in

Figure 3.

The static holds in the stretches were identified as the interval between the velocity spikes indicating transition periods. The specific frames at which the static hold began and ended were identified by a rule that set a number of consecutive frames required to be below the total standardized velocity threshold of 3.0. Any values lower would detect nonexistent transitions, while higher values fail to detect any transitions. Detailed examination of the data under different values for this number of frames indicated that for reliable identification of the hold period, four consecutive frames below threshold should be used to identify the start and three frames to identify the end. These numbers differ, because in the data, the release from a stretch tends to occur more quickly than the movement into the holding posture. The application of the rule to a single instance of the head joint during the Y stretch can be seen in Figure 4, where the green highlighted area represents the selected data.

V. Synthesized Feature Extraction

Once the period of static hold was identified, the multi-joint, 3D position data from frames in this period were used to extract features in the form of vector or scalar values determined by multiple joint locations (cf. on Gabel et al. [18]). Although raw 3D coordinates capture all the information in a movement, models trained with synthesized features yield higher accuracies [19]. As described in Table 3, eight synthetized features were computed, under the dual constraints of identifying informative elements of one or more poses, while discriminating between the different poses in the set of six targets. The full set of features was extracted from each individual frame that had been identified as static hold.

VI. Neural Network Evaluation on Synthetic Features

With the neural-net hyperparameters specified as described in Section 5, a classifier was constructed where the input to the model was now the synthesized features for each of the frames. To equate the contribution of the features, the values were normalized as z-scores across all the frames, pooling subjects and stretch categories. Frame content from the 60 exercise samples used previously was reduced by isolating only those frames that passed the threshold for normalized velocity associated with static holds. As before, for each run of the model, the 60 repetitions were divided into 42 used for training and 18 used for testing. The model was trained 20 times and its accuracies averaged to create the final performance metric. Figure 5 summarizes the entire data analysis flow to this point.

VII. Results

Table 4 shows the percentage of frames for each exercise that were classified at the .20 threshold and the ratio of responses with the given exercise name to the prevalence of that stimulus in the set. Ideal percentages of 100% would indicate that all frames that passed the threshold for static hold were classifiable, and that there was no response bias, i.e., the use of an exercise label matched its relative frequency in the data. The actual classification encompassed near or above 90% of the data. From the table, it can be seen that there was an evident response bias toward classifying frames as towel (relative to the ideal, +56%) and away from seated (-39%).

Table 5 presents the confusion matrix for the classification of individual frames, by exercise. Only responses exceeding

TABLE III
Features Synthesized from 3D Position Data

Feature (Vector Size)	Description	Calculations
Center of Mass (3)	Average position of all coordinates	$\frac{\sum_{n=1}^{25} X_{coord}, Y_{coord}, Z_{coord}}{25}$
Distance from Ankles to Wrist (1)	Average Euclidian distance between left wrist to left ankle and right wrist to right ankle	$\sqrt{(X_w - X_a)^2 + (Y_w - Y_a)^2 + (Z_w - Z_a)^2}$
Distance from Chest to Knee (1)	Euclidean distance of knee to chest averaged over left and right knees	$\sqrt{(X_k - X_c)^2 + (Y_k - Y_c)^2 + (Z_k - Z_c)^2}$
Orientation of Chest (1)	Stored as a 3D vector computed as the normal vector of two position vectors: the vector from the center of the chest to the left shoulder and the vector from the center of the chest to the right shoulder	$\overrightarrow{Vector1} = \overrightarrow{Chest} - \overrightarrow{Shoulder_{right}}$ $\overrightarrow{Vector2} = \overrightarrow{Chest} - \overrightarrow{Shoulder_{left}}$ $\overrightarrow{OrientationVector} = (\overrightarrow{Vector1}) \times (\overrightarrow{Vector2})$
Distance between Feet (1)	Euclidean distance from right ankle to left ankle	$\sqrt{(X_w - X_a)^2 + (Y_w - Y_a)^2 + (Z_w - Z_a)^2}$
Distance between Hands (1)	Euclidean distance from right wrist to left wrist	$\sqrt{(X_l - X_r)^2 + (Y_l - Y_r)^2 + (Z_l - Z_r)^2}$
Ankle between Feet (1)	Angle formed between the center of the hip, the right ankle and left ankle	$\overrightarrow{Vector1} = \overrightarrow{Hip_{center}} - \overrightarrow{Ankle_{right}}$ $\overrightarrow{Vector2} = \overrightarrow{Hip_{center}} - \overrightarrow{Ankle_{left}}$ $Angle = \arccos \frac{\overrightarrow{Vector1} \cdot \overrightarrow{Vector2}}{\|\overrightarrow{Vector1}\| \|\overrightarrow{Vector2}\|}$
Ankle between Hands (1)	Angle formed between the center of the hip, the right ankle and left ankle	$\overrightarrow{Vector1} = \overrightarrow{Chest_{center}} - \overrightarrow{Wrist_{right}}$ $\overrightarrow{Vector2} = \overrightarrow{Chest_{center}} - \overrightarrow{Wrist_{left}}$ $Angle = \arccos \frac{\overrightarrow{Vector1} \cdot \overrightarrow{Vector2}}{\|\overrightarrow{Vector1}\| \|\overrightarrow{Vector2}\|}$

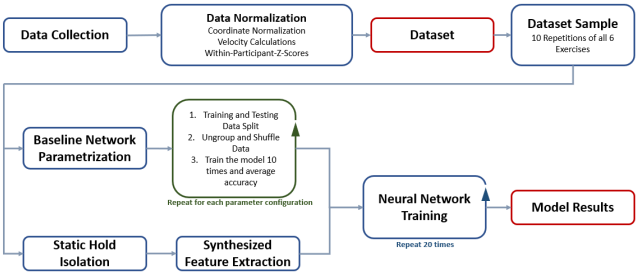

Fig. 5. Data Flowchart

TABLE IV
BY EXERCISE, % OF FRAMES PRESENTED THAT WERE CLASSIFIED AT THE .20 THRESHOLD, AND THE RATIO OF RESPONSES WITH THE GIVEN NAME TO ITS PREVALENCE IN THE DATA SET.

Exercise Name	% Classified	Response/Stimulus
Y	94.9%	107.5%
Sumo	95.8%	155.7%
Mermaid	91.3%	70.8%
Towel	87.1%	117.4%
Seated	90.9%	61.2%
Wall	98.9%	110.0%

TABLE V
ABOVE-THRESHOLD RESPONSE CLASSIFICATION FOR EACH STIMULUS AS A PERCENTAGE OF THE FRAMES PRESENTED FOR THAT STIMULUS. MARGINAL ENTRIES SHOW THE NUMBER OF ITEMS PRESENTED AND TOTAL NUMBER OF RESPONSES BY EXERCISE. DIAGONALS ARE CORRECT RESPONSES.

Exercise Presented	Response Exercise (>Threshold)						
	Y	Sumo	Mermaid	Towel	Seated	Wall	Stimulus N
Y	0.54	0.15	0.06	0.16	0.09	0.00	1351
Sumo	0.15	0.78	0.01	0.13	0.01	0.00	1288
Mermaid	0.20	0.09	0.52	0.14	0.05	0.02	1167
Towel	0.18	0.22	0.03	0.53	0.01	0.01	1212
Seated	0.00	0.21	0.04	0.12	0.50	0.01	1869
Wall	0.01	0.05	0.02	0.01	0.02	0.89	344
Response N	1452	2005	826	1423	1143	382	7231

confidence of .20 are included. The diagonals, indicating correct responses, ranged from .50 to .89, relative to a chance level of .17. Clearly, the performance was better than chance (the average of .63 was nearly a four-fold improvement) and better than the baseline of .25 established when all frames were classified on the measures from normalized camera coordinates. At a confidence of 0.20, a measure of information transmission in bits was .78 [20]. The maximum possible information transmission for a 6 X 6 matrix is 2.58 bits; however, a more appropriate benchmark would be human classification of the same frames, for which further data would be useful.

VIII. DISCUSSION

This paper describes development of a method of exercise classification using data attained by a Kinect depth camera, capitalizing on inter-frame and inter-joint dependencies. The approach provides a classification accuracy that is well above chance and far exceeds a network that attempts blind classification without capitalizing on temporal or spatial relationships. The essential steps are summarized in Figure 5.

There is reason to believe that the level of effectiveness of the current classifier could be improved without altering the approach. One limitation was the Kinect v2 camera used in this study, which was released in 2014. Since then, more powerful depth cameras have been developed that provide clearer and more consistent data. Future work could also benefit by capturing data from different perspectives to provide a multi-angle set. Current poses with the feet forward were particularly disadvantaged by the camera perspective. Multiple camera views could not only provide better position resolution but could yield more effective synthetic features. In addition, future work could explore different types of machine learning algorithms and deep learning systems. Further studies could use Keras or PyTorch to explore other deep learning techniques. Neural network hyperparameters developed for the final classifier, rather than the full-frame data set of joint positions, could result in improved accuracy.

In this study, we used our CHAR approach in conjunction with a synthetic feature SHAR model to improve single subject, single exercise classification. However, there remain numerous alternative applications in a conventional CHAR system, or in conjunction with other preexisting SHAR approaches. These novel approaches have potential to significantly impact model performance and warrant further research.

REFERENCES

[1] Y. Chang, S. Chen and J. Huang, "A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities", Research in Developmental Disabilities, vol. 32, no. 6, pp. 2566-2570, 2011.

[2] R. Ortiz-Gutiérrez, R. Cano-de-la-Cuerda, F. Galán-del-Río, I. Alguacil-Diego, D. Palacios-Ceña and J. Miangolarra-Page, "A Telerehabilitation Program Improves Postural Control in Multiple Sclerosis Patients: A Spanish Preliminary Study", International Journal of Environmental Research and Public Health, vol. 10, no. 11, pp. 5697-5710, 2013.

[3] A. A. M. Bigy, K. Banitsas, A. Badii, and J. Cosmas, "Recognition of Postures and Freezing of Gait in Parkinsons Disease Patients Using Microsoft Kinect Sensor", 7th Annu. Int. IEEE EMBS Conf. Neural Eng.,pp. 731734, 2015.

[4] K. Kaewplee, N. Khamsemanan, and C. Nattee, "A rule-based approach for improving Kinect Skeletal Tracking system with an application on standard Muay Thai maneuvers", 2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS), 2014.

[5] G. Zhu, L. Zhang, P. Shen and J. Song, "An Online Continuous Human Action Recognition Algorithm Based on the Kinect Sensor", Sensors, vol. 16, no. 2, p. 161, 2016.

[6] L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari and G. Serra, "Effective Codebooks for human action categorization", 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, 2009.

[7] M. Raptis and L. Sigal, "Poselet Key-Framing: A Model for Human Activity Recognition", 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[8] M.A. Gowayyed, M. Torki, M.E. Hussein, and M. El-Saban, "Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition", 23rd International Joint Conference on Artificial Intelligence, pp. 1351-1357, 2013.

[9] L. Xia, C. Chen and J. Aggarwal, "View invariant human action recognition using histograms of 3D joints", 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012.

[10] O. Oreifej and Z. Liu, "HON4D: Histogram of Oriented 4D Normals for Activity Recognition from Depth Sequences", 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[11] M.E. Hussein, M. Torki, M.A. Gowayyed, and M. El-Saban, "Human Action Recognition Using a Temporal Hierarchy Of Covariance Descriptors on 3D Joint Locations", 23rd International Joint Conference on Artificial Intelligence, pp. 2466-2472, 2013.

[12] X. Yang and Y. Tian, "Effective 3D action recognition using EigenJoints", Journal of Visual Communication and Image Representation, vol. 25, no. 1, pp. 2-11, 2014.

[13] P. Guo, Z. Miao, Y. Shen, W. Xu and D. Zhang, "Continuous human action recognition in real time", Multimedia Tools and Applications, vol. 68, no. 3, pp. 827-844, 2012.

[14] H. Eum, C. Yoon, H. Lee and M. Park, "Continuous Human Action Recognition Using Depth-MHI-HOG and a Spotter Model", Sensors, vol. 15, no. 3, pp. 5197-5227, 2015.

[15] A. Chaaraoui and F. Flórez-Revuelta, “Continuous Human Action Recognition in Ambient Assisted Living Scenarios”, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pp. 344-357, 2015.
[16] T. Sanchez Lengling, “Kinect v2 Processing library for Windows”, Codigogenerativo.com, 2018. [Online]. Available: http://codigogenerativo.com/kinectpv2/.
[17] “10 Stretches for Your Back”, Best Health Magazine Canada, 2018. [Online]. Available: https://www.besthealthmag.ca/best-you/stretching/10-stretches-for-your-back/.
[18] M. Gabel, R. Gilad-Bachrach, E. Renshaw and A. Schuster, “Full body gait analysis with Kinect”, 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2012.
[19] S. Maudsley-Barton, J. McPhee, A. Bukowski, D. Leightley and M. Yap,“A comparative study of the clinical use of motion analysis from Kinect skeleton data”, 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017.
[20] H. Tan, N. Durlach, C. Reed, W. Rabinowitz, Information transmission with a multifinger tactual display, Attention, Perception & Psychophysics, vol 61 (6), pp. 993-1008.