Skeleton data pre-processing for human pose recognition using Neural Network*

Bruna M.V. Guerra, Stefano Ramat, *Member, IEEE*, Roberto Gandolfi, Giorgio Beltrami and Micaela Schmid

***Abstract*— Automatic monitoring of daily living activities can greatly improve the possibility of living autonomously for frail individuals. Pose recognition based on skeleton tracking data is promising for identifying dangerous situations and trigger external intervention or other alarms, while avoiding privacy issues and the need for patient compliance. Here we present the benefits of pre-processing Kinect-recorded skeleton data to limit the several errors produced by the system when the subject is not in ideal tracking conditions. The accuracy of our two hidden layers MLP classifier improved from about 82% to over 92% in recognizing actors in four different poses: standing, sitting, lying and dangerous sitting.**

I. Introduction

Nowadays, the growing number of frail people (elderly or people affected by motor and/or cognitive disabilities) living alone in their own home is becoming a clinical and social problem. Their ability to perform daily tasks and to take care of themselves is severely impaired with the consequent increase of the risk of exposing themselves to dangerous situations.

In the last few years, a consistent technology effort was devoted to implement technological solutions aimed at increasing safety (i.e. Human Activity Recognition (HAR) for fall prevention and detection) and improving quality of life (i.e. automatic light switches, automatic doors and shades, motorized beds) of disabled and elderly people [1]–[3]. HAR identifies the daily activities of a monitored individual using information coming from wearable sensors or cameras. In the Ambient-Assisted Living (AAL) field HAR is widely used to automatically distinguish a normal behavior from a dangerous one to generate, when suitable, an alarm signal. In principle, the typical data analysis adopted in these solutions follows these steps: 1) raw data coming from wearable or environmental devices are processed and analyzed in order to select the most significant and informative features which best describe the stored data; 2) Artificial Intelligence techniques or threshold algorithms are used to define poses or daily activities performed by the subject [1], [4].

Computer vision-based systems offer a new, low-cost and promising solution for HAR, especially for healthcare monitoring of frail subjects. The devices commonly used for HAR and for monitoring purposes are depth cameras, such as Asus Xtion (Taipei, Taiwan), Intel RealSense (Santa Clara, USA), Orbbec Astra (Troy, USA) and Microsoft Kinect (Redmond, USA). The most common depth sensing device in AAL is the Microsoft Kinect sensor due to its affordable cost and support of custom monitoring and activity recognition software, which can be easily developed using its Software Development Kit (SDK) [1], [6]–[10].

The Kinect sensor has the advantage of being a non-intrusive device, which does not require the willingness of the user to wear it and is not limited by battery life. The resulting data is also respectful of privacy, thanks to the tracking of body joints computed from depth images, yet allowing 24h/7d subject's monitoring [7], [9]. Although between the first (Kinect v1) and the second version (Kinect v2) there have been hardware and software improvements [7], the Kinect v2 still has some drawbacks in the acquisition and processing of data, such as: 1) variable frame intervals [7]; 2) missing data when the subject is at the edge of the camera's calibrated volume or is partially hidden by furniture in the scene (es. desk, bed and chair [10]; 3) incorrect reconstruction of joints positions due to: noisy data, room lighting or overlapping of two or more joints [4], [11]–[14]; 4) recognition of 'ghost' skeletons caused by moving objects (i.e. chair) [10].

A classification model requires a reliable and validated dataset to efficiently generate the decision making rules. To improve the classification accuracy, it is important to evaluate the input data provided to the classifier and, if necessary, apply data pre-processing techniques to make them more reliable. Many studies using Kinect data for HAR in AAL proposed pre-processing algorithms with the goal of obtaining a reduction of erroneous information, misleading for the classifier [11], [14]. Sometimes, even the pre-processing is not enough to obtain an accurate classification of specific daily postures like lying or bending. Therefore, Li et al. (2019) proposed, following a data pre-processing stage, a hybrid approach running on anthropometric constraints together with a Neural Network classifier [13]. Nevertheless, the proposed method has limited performances when recognizing bending and lying postures in more directions and when the person sits laterally with respect to the Kinect v2. The aim of this work is therefore to define a Kinect v2 skeleton data pre-processing algorithm to partially overcome the limitations of the device mainly coming from its use in suboptimal conditions of viewing angles. This procedure hopefully can improve the performance of a Multi-Layer Perceptron (MLP) classifier proposed in a previous study by our group [15]. The MLP network with an average accuracy of 83,9% classified four poses assumed by an actor at varying orientations with respect to the camera (disadvantageous situation for skeleton

*Research supported by Regione Lombardia, project ID 379357, 2018.
B.M.V. Guerra is with the Computer, Electrical and Biomedical Engineering Department, University of Pavia, Pavia, 27100 Italy (corresponding author to provide -mail:brunamariavitt.guerra01@universitadipavia.it).

S. Ramat, R. Gandolfi, G. Beltrami and M. Schmid are with the Computer, Electrical and Biomedical Engineering Department, University of Pavia, Pavia, 27100 Italy (e-mail: stefano.ramat@unipv.it, robgan08@unipv.it, giorgio.beltrami@unipv.it, micaela.schmid@unipv.it).

reconstruction): standing, sitting, lying down and dangerous sitting. The latter consisted of the subject slumped in a chair.

II. METHODS

A. Subjects

11 normal subjects (7 females and 4 males; age ranging 25 and 60 years old; height ranging 1.55 and 1.90 m) participated in the study. All subjects gave written informed consent in accordance with the Declaration of Helsinki.

B. Instrumentation and Acquisitions

Microsoft Kinect is a low-cost motion sensing instrument, initially used as an input device for the Microsoft Xbox gaming console, then extended to many fields. Thanks to its hardware design and SDK it is possible to detect, track and recognize human motion in real time [8]. The Kinect v2 system can detect a human body and voice signal using a Full HD RGB camera, a depth sensor and an array of four microphones. Its nominal frame rate is 30 Hz and the viewing angle is 60° vertically and 70° horizontally. It is capable of tracking 25 joints for up to 6 actors simultaneously. The ideal experimental setup requires the subject in front of the sensor at a distance ranging of 0.8 - 3.5 m.

Experimental acquisitions were performed in a prototype bedroom. Subjects were asked to perform four poses: standing, sitting, lying down and dangerous sitting. The last one grouped all situations representing malaise or fainting and resulting in a seated person slumped or lying backwards. Actually, these postures are kinematically different but we chose to collapse them in a single pose because our focus is to classify a dangerous situation (fainting) and not to detail the specific posture assumed by the subject in this circumstance. The acquisitions, lasting about 13 min per subject, were structured as follows:

- the subject starts to walk form standing position then grabs a chair near the desk and sits on it. While sitting, he first moves the head backwards and then leans the trunk forward while simultaneously pitching the head as an unconscious person (dangerous sitting). The subject then returns to the normal sitting position and finally gets up and brings the chair back to its original location (standing).
  Each pose was maintained for 10 seconds. This sequence of poses was repeated four times, each one performed in different room locations and subject orientations and stored in four separate recordings.
- the subject starts sitting on the bed, then lies down on the back and turns to the right side. The subject then returns on the back and turns to the other side. The sequence was recorded four times.

The sequence of poses in each acquisition was timed by the operator running the experiment.

C. Data Processing

1) Skeletal Tracking: Using the Microsoft SDK 2.0, we computed the spatial coordinates (x, y, z) of the standardized 25 skeletal joints. We reduced the number of joints to obtain a minor but reasonable set of joints which are the most involved in the poses of our interest. An additional joint Hc (17 in Fig. 1) was computed as the midpoint between the two hips joints. All 17 analyzed joints are shown in Fig. 1. Then, the coordinates of the 17 joints were roto-translated to obtain data referred to an absolute reference system (X, Y, Z) space-fixed in the room [15].

2) Pre-processing algorithms: We propose two pre-processing algorithms, both based on several thresholding procedures aimed at removing unlikely data, but differing for the addition, in the second one, of: a) a linear fitting method aimed at approximating the missing or out-of-threshold data; b) data averaging over a temporal sequence of 15 frames (corresponding to 0.5 seconds).

In the first pre-processing algorithm, raw data are averaged over temporal windows of 15 frames and a threshold corresponding to the mean ± 3 Standard Deviations (SD) is applied to detect and remove outliers. All data overcoming such threshold are removed. If a time window contains more than 30% of missing data, the data frame is deleted. After such data cleaning, body segment length is computed for each pair of consecutive joints and then compared with the corresponding anthropometric value for a normal subject [16]. Since the joint positions, calculated by the anthropometric reference model, do not exactly correspond to those processed by Kinect v2, a tolerance threshold of 40% is considered. This, as well as the following controls, is carried out only on the subset of joints considered more accurate [17], namely: head (1), C7 (2), acromion (3 - 4), iliac crest (9 - 10), Hc (17) (Fig. 1). In addition, a tolerance of only 30% on the variability of each body segment length between consecutive frames is considered acceptable. Finally, the velocity of joints movements between two successive frames is taken into account. Assuming that the velocity of a subject during a natural walking pace ranges between 4 and 5 km/h, the threshold is set to 6.48 km/h (displacement of 6 cm between two frames). If one of the comparisons described above does not meet the threshold condition, all data referring to that frame is removed. Since the purpose of this pre-processing algorithm was to provide reliable data input to a neural MLP network, the choice of all the threshold values was made to ensure a good compromise between quality and quantity of data.

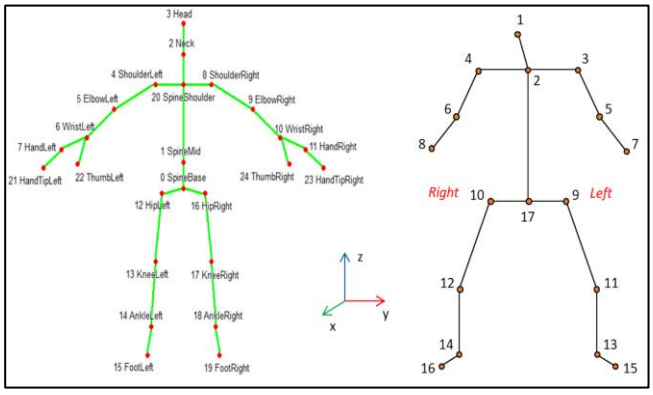

Figure 1. The 25 joints skeleton computed by SDK (on the left) and the reduced 17 joints skeleton used for the analysis (on the right).

In the second pre-processing algorithm to reduce the temporal discontinuity of the raw data, we perform an approximation of the missing data by a linear fitting. The values of the 4 frames preceding and following the missing data are used for the approximation, which is made only if at least five valid values are found around the missing one. Alternatively, no replacement is made. Then, the processed data are analyzed with the same mean ± 3 SD threshold procedure described in the first pre-processing algorithm and the removed samples are fitted again with the linear procedure described above. Finally, data are further handled with an averaging procedure over a time window of 15 frames. This process limits the frequency of the MLP classification at 2 Hz, but we consider that it may be enough to recognize dangerous situations in the AAL domain. The following steps of this pre-processing algorithm replicate those of the first one.

D. Neural Network

1) Kinematic Features Definition: A set of kinematic features that can characterize different human poses independently of each subject's body size was defined, namely: vertical position of the head, C7 and Hc joints (1, 2, 17 in Fig. 1) each normalized with respect to the subject's height, three relative angles between two consecutive body segments (head-shoulder axis, head-trunk, trunk-iliac crest axis) and the absolute roll and pitch angles of the head and trunk. All angles were normalized by dividing them by 180°.

2) Databases: We defined four classes as follows:

- Class1: standing pose;
- Class2: sitting pose;
- Class3: lying down pose;
- Class4: dangerous sitting pose.

Since we used a supervised neural network, we labelled each frame with the corresponding class using a custom-made software (MATLAB 2019a), which also allowed us to identify the frames corresponding to the transition from a pose to another and to remove them from the dataset. This cutting procedure is suitable in this context because the goal of the neural network algorithm is that of recognizing a pose when it is accomplished, in a static condition (i.e. subject lying down on the floor and not during the fall) [15].

The raw data (434264 frames), the first algorithm pre-processed data (77562 frames) and the second algorithm pre-processed data (24484 frames) were used to generate the training and test databases. The first was built on 7 subjects and the second one on 4 subjects. Finally, we have trained and tested the network applying the following data combinations:

- raw data for training (282820 frames) and testing (151444 frames) (*Case A*);
- first algorithm pre-processing data for training (49530 frames) and testing (28032 frames) (*Case B*);
- second algorithm pre-processing data for training (15664 frames) and testing (8820 frames) (*Case C*);
- first algorithm pre-processing data for training (49530 frames) and second algorithm pre-processing data for testing (8820 frames) (*Case D*).

3) MLP: MLP Neural Network was implemented in MATLAB 2019a using Neural Network Toolbox. The proposed network consists in an input layer connected to the 10 features describing each frame in the database, two hidden layers and an output layer having a 'SoftMax' transfer function. The MLP network was trained using the Levenberg-Marquardt backpropagation algorithm, first with a k-fold cross validation (k=10), and then using the whole training set. The learning process was performed over a maximum of 1000 epochs, i.e. 1000 iterations on the training set. Precision, sensitivity, specificity and F-score were calculated for each fold and then the same parameters were computed over the 10 folds (mean values). The trained MLP network was then tested over the whole test database and the accuracy was computed (TABLE I). Considering these results, we further investigated the data by computing class precision, sensitivity, specificity and F-score in the *Case A* (control scenario) and *Case D* (best accuracy scenario).

III. RESULTS

Fig. 2 shows an example of the head joint vertical position raw data (upper panel), characterized by noise and temporal hole (missing data), and the effects of the pre-processing algorithms on these data (middle and bottom panel). The first pre-processing erases the noisy data which do not satisfy the criteria defined in the algorithm. All the three coordinates of each joint are analyzed, and it is enough that a single data coordinate exceeds the threshold to determine the deletion of the whole frame. For this reason, as shown in Fig. 2 (middle panel), some raw data that appear stable for head Z coordinate are nonetheless removed. The second pre-processing algorithm (bottom panel), in addition to eliminating noise on the raw data, gives the data temporal regularity thanks to the fitting and the averaging procedures.

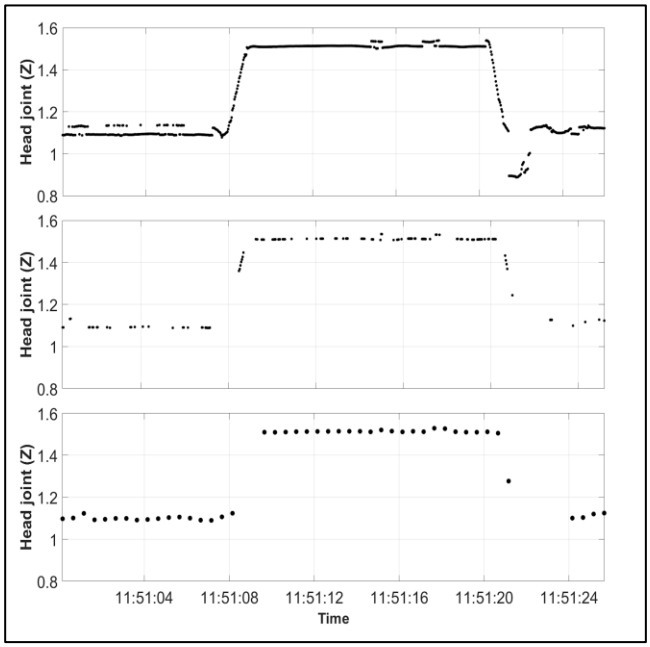

Figure 2. Example of data head vertical position-trace: raw data (top panel); first algorithm pre-processed data (middle panel); second algorithm pre-processed data (bottom panel).

TABLE I. MLP NEURAL NETWORK ACCURACY

Case A	*Case B*	*Case C*	*Case D*
86.5	91.6	91.9	94.5

TABLE II. CLASSIFICATION RESULTS FOR *CASE A* AND *D*

	Precision (%)		**Sensitivity (%)**		**Specificity (%)**		**F-score (%)**	
	Case A	***Case D***	***Case A***	***Case D***	***Case A***	***Case D***	***Case A***	***Case D***
Class1	95.4	96.1	92.3	98.1	98.1	98.2	93.8	97.1
Class2	88.6	96.8	90.3	94.0	90.5	97.2	89.5	95.4
Class3	67.5	88.9	75.9	86.6	96.8	99.5	71.4	87.7
Class4	75.8	87.8	72.9	91.7	94.2	97.0	74.3	89.7
Mean value	**81.8**	**92.4**	**82.8**	**92.6**	**94.9**	**98.0**	**82.3**	**92.5**

As shown in TABLE I, all MLP accuracy results the pre-processed data are higher than on the raw data. The best accuracy was found in *Case D* (first algorithm pre-processing data for training and second algorithm pre-processing data for testing). The statistical results referring to each class for *Case A* (control scenario) and *Case D* (best accuracy scenario) are summarized in TABLE II. These results confirm the classification improvements obtained when pre-processing data.

IV. DISCUSSION AND CONCLUSION

In this work a pre-processing procedure for Kinect v2 data is proposed. The aim was to provide more stable and reliable data as input to a two hidden layers MLP neural network for pose classification, in order to improve its performance. The data were acquired during the monitoring of a subject performing daily living activities inside a room. Two algorithms were defined. The first removes the data which do not satisfy different threshold criteria based on variability, anthropometric measures and joints velocity. The second one mimics the first one, but additionally reconstructs the missing data and averages the data on a half-second time window. In order to verify the effectiveness of the data processing on MLP network classification, we compared the results obtained with different combinations of training and testing pre-processed data with those resulting using the raw one. The accuracy of the MLP network showed that network performance improves in all cases in which the data are pre-processed. The best performance seems to be obtained by training the network with data processed by the first pre-processing algorithm and to test it with data calculated by the second pre-processing algorithm (*Case D*). This is probably due to the fact that in *Case D* the advantages of each of the two procedures are exploited. The first algorithm, used for the training data, retains a good cardinality and hence diversity of examples, despite deleting the noisy data. The second algorithm, used for data testing, provides an increased regularity of the data, although reducing its cardinality. Moreover, this method allows to increase considerably the correct classification of two particularly relevant classes for automatic dangerous situations recognition systems in AAL, i.e. lying down and dangerous sitting poses. This occurs despite the subject orientation with respect to the camera.

In order to discriminate lying pose normally assumed during the day from dangerous situations, a foreseen further development step will be to integrate the Kinect v2 data with that acquired from a network of sensors placed in the room. Furthermore, the PROPOSED method PROVIDES good results in the off-line analysis, but considering that the use case will be in real-time, new tests and validations will HAVE TO be carried out in this condition.

REFERENCES

[1] S. Ranasinghe, F. Al MacHot, and H. C. Mayr, "A review on applications of activity recognition systems with regard to performance and evaluation," *Int. J. Distrib. Sens. Networks*, vol. 12, no. 8, pp. 1-22, 2016.

[2] V. Bevilacqua, N. Nuzzolese, D. Barone, M, Pantaleo, M. Suma and D. D'Ambruoso, "Fall detection in indoor environment with kinect sensor," *INISTA 2014 - IEEE Int. Symp. Innov. Intell. Syst. Appl. Proc.*, pp. 319–324, 2014.

[3] Z. Mundher, A. Zaid and J. Zhong a, "A Real-Time Fall Detection System in Elderly Care Using Mobile Robot and Kinect Sensor," *International Journal of Materials, Mechanics and Manufacturing*, vol. 2, no. 2, pp. 133–138, 2014.

[4] S. M. M. Ali, J. C. Augusto, and D. Windridge, "A Survey of User-Centred Approaches for Smart Home Transfer Learning and New User Home Automation Adaptation," *Appl. Artif. Intell.*, vol. 33, no. 8, pp. 747–774, 2019.

[5] G. Mastorakis and D. Makris, "Fall detection system using Kinect's infrared sensor," *J. Real-Time Image Process.*, vol. 9, no. 4, pp. 635–646, 2014.

[6] R. A. Clark, B. F. Mentiplay, E. Hough, and Y. H. Pua, "Three-dimensional cameras and skeleton pose tracking for physical function assessment: A review of uses, validity, current developments and Kinect alternatives," *Gait Posture*, vol. 68, pp. 193–200, 2019.

[7] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with Microsoft Kinect sensor: A review," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1318–1334, 2013.

[8] R. A. Clark, Y. Pua, K. Fortin, C. Ritchie, Webster K. E., Denehy E., A.L. Bryant, "Validity of the Microsoft Kinect for assessment of postural control," *Gait Posture*, vol. 36, no. 3, pp. 372–377, 2012.

[9] C. Rougier, E. Auvinet, J. Rousseau, M. Mignotte, and J. Meunier, "Fall detection from depth map video sequences," *International conference on smart homes and health telematics*., Springer, Berlin, Heidelberg, pp. 121-128, 2011.

[10] M. S. Alzahrani, S. K. Jarraya, H. Ben-Abdallah, and M. S. Ali, "Comprehensive evaluation of skeleton features-based fall detection from Microsoft Kinect v2," *Signal, Image Video Process.*, vol. 13, no. 7, pp. 1431–1439, 2019.

[11] R. Sahak, N. K. Zakaria, N. M. Tahir, A. I. M. Yassin, and R. Jailani, "Review on Current Methods of Gait Analysis and Recognition using Kinect," *Proc. - 2019 IEEE 15th Int. Colloq. Signal Process. its Appl. CSPA 2019*, no. March, pp. 229–234, 2019.

[12] B. Li, C. Han, and B. Bai, "Hybrid approach for human posture recognition using anthropometry and BP neural network based on Kinect V2," *Eurasip J. Image Video Process.*, vol. 2019, no. 1, 2019.

[13] Z. P. Bian, J. Hou, L. P. Chau, and N. Magnenat-Thalmann, "Fall detection based on body part tracking using a depth camera," *IEEE J. Biomed. Heal. Informatics*, vol. 19, no. 2, pp. 430–439, 2015.

[14] R. Li, W. Si, M. Weinmann, and R. Klein, "Constraint-based optimized human skeleton extraction from single-depth camera," *Sensors (Switzerland)*, vol. 19, no. 11, pp. 1–20, 2019.

[15] B. M. V. Guerra, S. Ramat, G. Beltrami, and M. Schmid, "Automatic pose recognition for monitoring dangerous situations in Ambient-Assisted Living," *Front. Bioeng. Biotechnol.*, vol. 8, p. 415, 2020.

[16] D. A Winter, "Biomechanics and motor control of human movement". *John Wiley & Sons*, 2009 .

[17] M. Wochatz, N. Tilgner, Mueller S., Rabe S., Eichler S., Jhon M., Völler H., Mayer F, "Reliability and validity of the Kinect V2 for the assessment of lower extremity rehabilitation exercises," *Gait Posture*, vol. 70, pp. 330–335, 2019.