Recognition of Human-Object Interaction by means of wearable sensors' originated signals

A. Di Tecco, A. Le Caldare, A. Liuzzi, R. Mule', L. Porfilio, F. Scotto, L. Treccozzi, A. Vecchio

***Abstract*—In this work a neural-network based interaction-detector is proposed. On the basis of the signals coming from two pairs of accelerometers and gyroscopes (placed, respectively, on the user's wrist and inside a backpack) an interaction detection with a high precision can be performed. Results have shown that, with the proper choice of features, it is possible to detect an interaction with an accuracy of roughly 99%.**

***Index Terms*—Human-Object interaction detection, Neural Networks, Trajectories, MEMs, accelerometer, gyroscope, wearable sensors.**

I. Introduction

Wearable devices are daily used by an always increasing fraction of people. Smart-wristbands and smart-watches provide an easy way to monitor the activity level of users, to log training sessions, and to unobtrusively receive notifications [1], [2]. Smart-shoes, initially adopted in the e-health domain, are now getting wider acceptance from athletes and sport enthusiasts, as a means to obtain detailed information about their running parameters. Also smartphones, at least intermittently, may be somehow included in the wearable category, as they are frequently carried by their owners in a pocket. Data originated by wearable devices has been extensively used for inferring the context of the user [3]. In particular, a large amount of literature focuses on how users' activities can be recognized from information provided by accelerometers, gyroscopes, and other sensors attached to their bodies [4], [5], [6].

Significantly less attention has been dedicated at recognizing the interaction between users and smart-objects available in the environment. This work aims to set-up an interaction-detector that, provided signals from sensors (namely a MEM accelerometer and a MEM gyroscope), instantly fires when the user is interacting with the smart-object.

II. Related Work

A. Approximate string matching: A lightweight approach to recognize gestures with Kinect

This paper [?] describes a method which uses string similarity algorithms to classify body gestures.
The work is based on the Microsoft Kinect platform, in particular on skeleton data (joints position/trajectory) generated by the device. The procedure is based on 2 phases, training and detection. During the training phase a dataset of gestures is built: one/more people try to make many gestures in different ways. The goal is to build a template for each gesture, which will be used during the detection phase. First, gestures are processed to be comparable (independent from camera position and body size), then a k-means clustering algorithm is used to obtain k centroids from all the samples of a particular gesture. A unique char is assigned to each cluster and all gesture samples are encoded as strings, assigning each trajectory point to the nearest centroid. A string similarity algorithm is used to select the most similar/least-distant gesture and a threshold/tolerance. During the detection phase, a new gesture is processed and classified similarly to the last training steps: for each template gesture the distance between the template string and the new string is computed. Distance/Tolerance is used to classify the new gesture.

B. Pick Me Up and I Will Tell You Who You Are: Analyzing Pick-Up Motions to Authenticate Users

Reducing user active efforts as much as possible for authentication, the researchers in want to describe in [7] how to authenticate a subject who picks up a device, e.g. its smartphone. The researchers got the authentication of a person with an accuracy of 85%. Then they studied 24 subjects (14 men, 8 women), whose age ranged between 19 and 62 years, with a median of 27 years old. All participants are ambidextrous. The height of the participants varies between 153 and 198 cm. During the collection phase, an accelerometer, a gyroscope and a data rotation sensors were used. These sensors were installed on a Nexus 6 smartphone that was used to gather data. The most important data gathered were average and variance of each sensors, for each axis. A multilayered perceptron network was used to classify subjects from each others, and this classifier has been validated with a leave-one-out cross-validation using WEKA. The experiment was conducted in a room with a fixed desk and chair. On the desk there were a dice and the smartphone. The latter was set up to record all the information with the apps mentioned above during the experiment. On the desk there were also two regions marked as A and B. One of these two regions represents the initial position of the smartphone. The choice of the region is based on the starting position of the subject: seated on the chair (A) or standing up next to the desk (B). In each experiment, the phone is taken in hand for a certain number of seconds equal to the number randomly generated by the dice; at the end of the experiment, the subjects are invited to put it back in the initial position. Each participant repeated 20 motions collection both in position A and B (40 per subject in total).

C. An Effectiveness Study on Trajectory Similarity Measures

GPS sensors are nowadays pervasive, we have data coming from thousands of devices. One important application for

Department of Electrical and Computer Engineering, University of Pisa, Pisa, 56126, Italy, e-mail: ing.unipi.it

Manuscript received May 28, 2019

those data is Trajectory Data Management: analyzing large amount of data describing the motion history of objects and say how much they are similar with respect to each other. This paper [8] focuses on six widely used trajectory similarity measures and provides an effectiveness study. This work is based on two observations: the same motion history can be represented by different trajectories; those representations should still have high similarity (based on any good measure). The major problem is the lack of benchmarks, so it is necessary to develop a set of transformation functions (increasing/decreasing sampling rate, random/synchronized point shifting, adding noise) for the original data. The transformed trajectories will be then compared with the original ones. Each transformation will be controlled using two parameters: rate (percentage of "victim points") and distance (how far the point should be transformed). Experiments were performed for each combinations of transformation and similarity measure. As result, no measure actually outperforms the others, but the effectiveness depends on the type of applied transformation.

D. SmartStuff: A Case Study of a Smart Water Bottle

This paper [9] describes a case study of a smart water bottle. The authors used several sensors in order to make the bottle "smart". For example, they used a custom developed Touch and Pulse Sensor (TAPS) to detect when the user touches the bottle, a photoplethysmographic sensor (PPG) and inertial sensors. The "touch detection" was possible thanks to the capacitive sensing of the TAPS and it is useful for determining if the user is using the bottle (useful for example to turn off some sensor when the bottle is not used). Furthermore, they also used a 3D accelerometer to detect the use of the bottle to identify the user. Combining the data coming from the inertial sensors of the bottle and from the user's smartwatch is possible to identify the user using the detection of the same pattern of activity; in fact, in case of interaction with the bottle, the acceleration measured by the smartwatch and the bottle is very similar if the bottle is held on the smartwatch hand, otherwise it is possible to detect the heart rate (from the PPG and from the smartwatch) for the user identification.

E. Feature Selection and Activity Recognition System Using a Single Tri-axial Accelerometer

This paper [10] describes an activity recognition system utilizing a waist-mounted accelerometer to classify six daily living activities. Data was collected on two young people that perform six activities in a room. The data signals are segmented into windows of six seconds and an overlap of 50% between two consecutive windows. A large set of features is extracted including some other new features that are insensitive to accelerometer position, in addition to features already used in literature like mean and energy. These features are Mean Trend and Windowed Mean Difference, Variance Trend and Windowed Variance Difference, Detrended Fluctuation Analyses coefficient and Uncorrelated X-Z Energy. Two separate classifiers are utilized: k-NN and Naïve Bayes. Feature selection was performed with a Filter based approach (Relief-F) that selects most relevant features that provide best re-substitution accuracy with the classifier, and a Wrapper based approach (SFFS), that picks in a subset of selected features the one that not increase the re-substitution error. The second one provides much better estimates for re-substitution error at less than half the number of features. It is interesting to note that the Mean Trend, Windowed Mean Difference and DFA coefficients were included as relevant features by both the classifiers. The feature validation and, therefore, activity recognition was performed on data from 7 subjects with the leave-one-person out strategy for train and classify activities. As result, the overall accuracy of the system was about 98% from both the classifiers. Wrapper based feature selection might provide better results than the filter based approach. New features provide good results for activity classification and were chosen ahead of other popular features by feature selection algorithms.

III. Method

A. Experiments set-up and data acquisition

For the human-object interaction study one person (the user, henceforth) and one backpack for each scenario were engaged. We have defined 4 scenarios, designated as scenario A, B, C and D below, involving 3 people who have repeated 5 tests for each scenario. The only object in the created room is a table, used as a support for the backpack.

Scenario A requires the user to enter the room without the backpack, move close to the table, grab the backpack from the table and leave the room with the backpack. Scenario B requires the user to enter the room with the backpack, move close to the table, leave the backpack on the table and leave the room without the backpack. Scenario C requires the user to enter the room with the backpack, move close to the table, place the backpack on the table, take something from the backpack and leave the room without the backpack. Scenario D requires the user to enter the room without the backpack, grab the backpack from the table, carry and place the backpack in a corner of the room and leave the room without the backpack.

In order to acquire data needed for our work we used the position sensors provided by a MDEK1001 kit by Decawave and the accelerometer and the gyroscope of two Shimmer3 IMU units. Regarding the MDEK1001 kit, four DWM1001-DEV boards were configured as anchor nodes in order to create a 3D-space virtual room in which to test our scenarios; two of them were configured as tag nodes and one as listener node. The anchor nodes have been placed on two parallel walls at the distance of 2.60m to each other and at the height of 2m from the floor. One of the tag nodes was positioned inside the backpack and the other one on the user wrist. The listener node has been connected to a laptop to which tag nodes stream acquired data. Fig. 1 shows the MDEK kit configuration and the technologies used by the nodes to exchange data. One Shimmer device was placed inside the backpack, the other was fastened to the tester wrist by means of a wrist strap. Data provided by the motion sensor of the MDEK kit were captured real-time by means of the listener node connected to the laptop. For each person, at the end of each scenario, data have been stored on the laptop.

As example, in Fig. 2, the L2-norm of the triaxial accelerometer, triaxial gyroscope and interdistance are reported from the sensor attached to the user's wrist and inside the backpack.

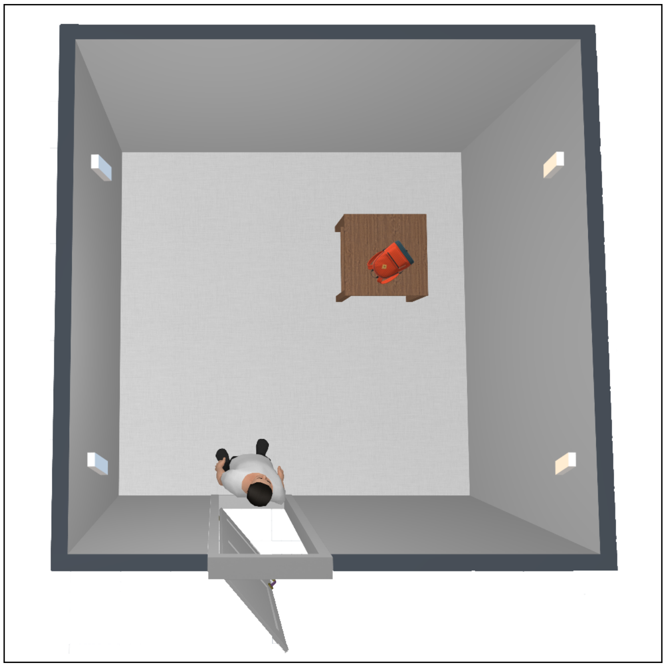

Fig. 1: Room created by means of MDEK position sensors.

B. Data cleaning

Once data have been captured, a data cleaning procedure must be performed as the acquired data may lack of some samples (due to the inherently unreliability of wireless channel over which they have travelled). To tackle with this problem, missing samples have been filled with linearly interpolated values. Furthermore, signals appear to be unsynchronized, hence additional effort is required in order to eliminate the incorrect displacement between them. To overcome this problem a simple solution has been adopted: every user has been asked to jump prior performing the sample interaction. This simple solution allowed us to detect a known starting point (identified by the high value acquired at that instant by the accelerometer and gyroscope) that has been then labelled to facilitate a further automatic labelling, and thus, synchronized the samples. The signal thus obtained have been smoothed in order to cut off possible noise. To this purpose a low-pass filter has been applied.

C. Derived signals

Additional signals have been created starting from those. There are two reasons for which it is preferable to follow this approach. The first one is that the signals are axis-dependent: a signal pattern along an axis depends both on the direction over which the interaction is being performed (same actions can lead to different signals along the three axis) and on the orientation of the sensors involved in the interaction. The solution is straightforward: rather than the raw per-axis signals, it would be more effective to also consider the L2-norm. The second reason to use additional signals is that some properties are *hidden* in their raw form: as an example, we extracted, from both the L2-norm and original signals, their cross-correlation and first-order derivative. Finally, from the position sensors, for the same reasons formerly discussed, we extracted the per-sample inter-distance.

D. Features extraction

Every pair of user-backpack signals has been split in a 1-second window with no overlapping samples as depicted in Fig. 2. For our experiments, every signal was composed on average by 15 windows, thus gathering, at the end, 928 windows. From each of them (user's and backpack's) and for each signals (no distinction will be made from original signals and derived signals from now on) 10 features have been extracted: minimum, maximum, mean value, variance, standard deviation, mean trend, energy and interquartile range. Every windows has been then labelled as interaction if it was inside the interaction window, that has been manually built by analyzing the video in which every user performs the interaction, and non-interaction otherwise. To label the window at the edges of the interaction (i.e. the windows in which the interaction is not started yet and those in which the interaction is not still concluded) an heuristic yet effective method has been adopted in order to further improve the detection accuracy. This method consists of splitting each window in as many quanta as the number of samples that this latter carries, and counting the number of quanta inside the interaction window. Let us call this counting-result q. If q is greater than 40% of the overall number of samples in the current windows, then this latter is labelled as interaction, otherwise as non-interaction.

IV. Classification using Intelligent System

We found the optimum features set in the iqr of the angular velocity user-backpack cross-correlation, maximum of angular acceleration of backpack, mean of angular velocity of backpack over all the 400 features per windows, by means of the Forward Sequential Feature Selection. We then applied statistical analysis techniques to find the optimal number of hidden neurons. We used the back-propagation algorithm to train the network. At the end we applied 10-fold cross validation. Lastly, the Student's t-test has been performed. The statistical test compares error distributions statistically and allows to determine the optimal number of neurons in the hidden layer. Once the number of optimal neurons $h*$ has been derived, we repeated what has been done to find the optimal network.

V. Results

Conducted experiments have led to an optimal accuracy in terms of interaction classification, allowing a clear distinction from the non-interaction class. By means of a neural network with a single hidden layer composed of three neurons it was possible to obtain a maximum accuracy of 99% and an average

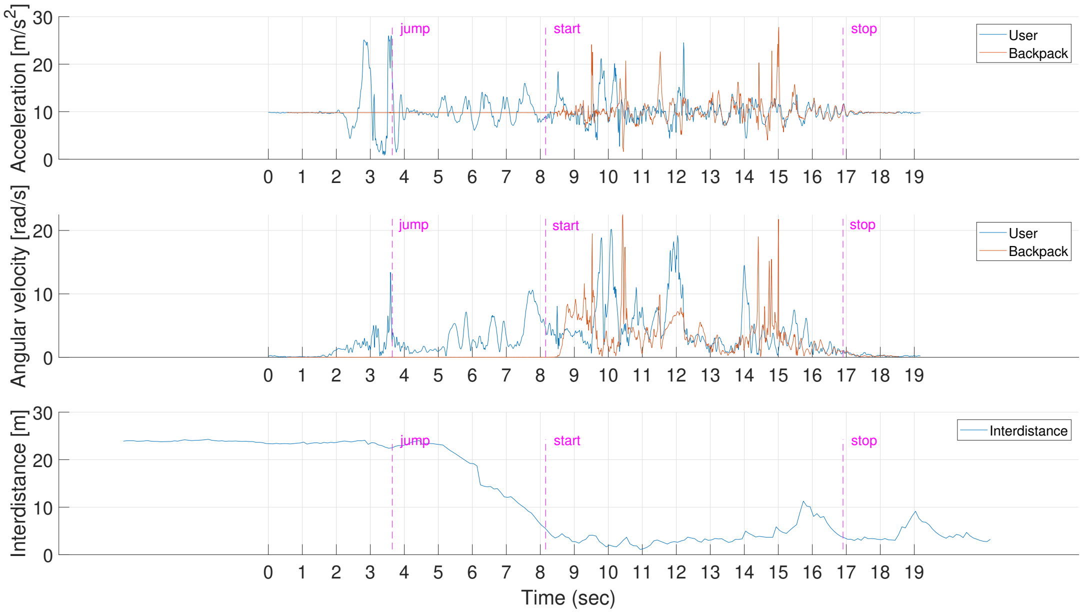

Fig. 2: The first plot shows acceleration of wrist's user and backpack. In the second is shown angular velocity of couple wrist-backpack. In the last chart is possible to see the interdistance between the user and the backpack.

accuracy of 98%, using the mean square error on the test set as evaluation metric.

In the following, confusion matrices (Fig. 3), error histogram (Fig. 4) and output classes distributions (Fig. 5-6) are depicted.

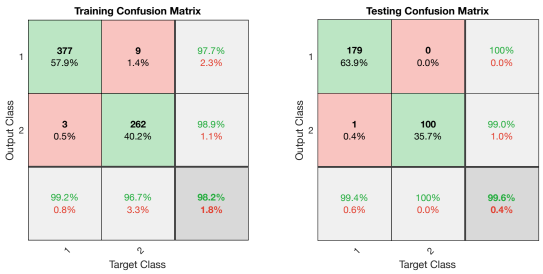

Fig. 3: Confusion matrices on training set and test set.

VI. Conclusion

Other classification models have been investigated: KNN, SVM, LD, LR, tree, bagged trees and rusboosted trees, all of which have been validated with 10-fold cross-validation. All models confirmed the previous results with an accuracy greater than 95%.

Analyzing the misclassified samples, we realized that the network corrected the windows of some tests incorrectly labelled (because of automatic labelling), as shown in Fig. 7. The correction of desired output associated with the windows of the wrongly labelled samples (windows at the edges of the interaction), would lead to an accuracy that is very close to 100%.

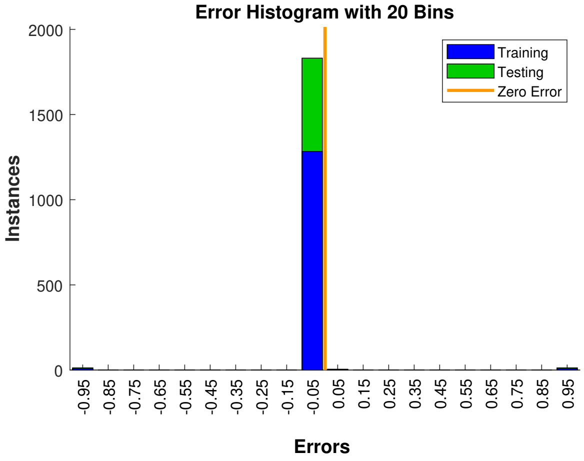

Fig. 4: Error distribution of classification.

We finally observed that, using only signals coming from the Shimmer sensors and extracting a greater number of features, it is possible to achieve a quite high accuracy, comparable to that obtained using signal coming from both devices. This model has the advantages that, provided a greater number of features as input, it does not depend by any anchor sensors, or, in other words, on the virtual room in which experiments have been conducted (see Table I). As expected, classifier accuracy using both interdistance and accelerometer/gyroscope-originated data is pretty high.

Now let's analyze an additional scenario: a person keeps the bag on his shoulders without moving, then leaves it on a table and seats somewhere far from the bag. This leads to an almost 0 acceleration and angular velocity, which makes

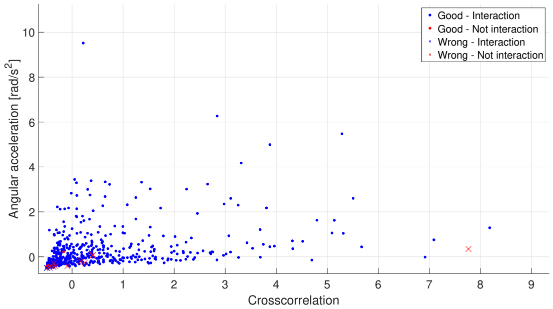

Fig. 5: Distribution of the classified samples in the scatter plot. On the X axis there is the cross-correlation of the angular velocity of the user-backpack, while on the Y axis the derivative of the angular velocity. Samples that are correctly classified during training and using test set are labeled as spots. Wrong classified samples are labeled as crosses.

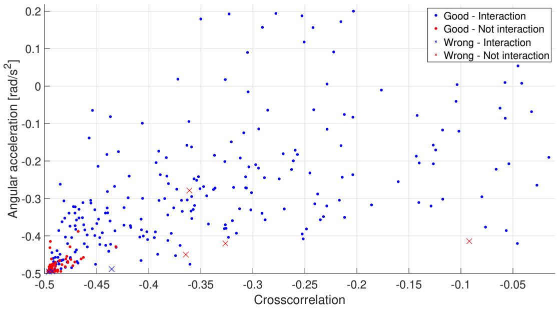

Fig. 6: Zoomed scatter plot in the region of non-interaction class.

the classification impossible without using interdistance as a feature. However, dropping the interdistance from the features list does not lead to a lower accuracy value, as earlier discussed. This means that our dataset does not cover all cases and thus needs to be expanded. One possible scenario that would expand the dataset could be the proposed one. For this scope, further scenarios may be added, and they could take into considerations:

- shorter/longer walking distance;
- different sensor body position and/or orientation;
- different model of bag and sensor position/orientation;
- slopes and stairs.

TABLE I: 95th percentile mean and variance building 36 networks using training and testing set for each MDEK, SHIMMER and MDEK+SHIMMER feature set.

	MDEK	**SHIMMER**	**MDEK+SHIMMER**
Mean	95.69	98.33	98.14
Standard Deviation	0.77	0.36	0.64

VII. Acknowledgment

The authors would like to thank...

References

[1] Y. Zheng, X. Ding, C. C. Y. Poon, B. P. L. Lo, H. Zhang, X. Zhou, G. Yang, N. Zhao, and Y. Zhang, "Unobtrusive sensing and wearable devices for health informatics," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 5, pp. 1538–1554, May 2014.

[2] R. Chambers, T. J. Gabbett, M. H. Cole, and A. Beard, "The use of wearable microsensors to quantify sport-specific movements," *Sports Medicine*, vol. 45, no. 7, pp. 1065–1081, Jul 2015. [Online]. Available: https://doi.org/10.1007/s40279-015-0332-9

[3] B. Clarkson, K. Mase, and A. Pentland, "Recognizing user context via wearable sensors," in *Digest of Papers. Fourth International Symposium on Wearable Computers*, Oct 2000, pp. 69–75.

[4] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1192–1209, Third 2013.

[5] S. Khalifa, G. Lan, M. Hassan, A. Seneviratne, and S. K. Das, "Harke: Human activity recognition from kinetic energy harvesting data in wearable devices," *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1353–1368, June 2018.

[6] P. Casale, O. Pujol, and P. Radeva, "Human activity recognition from accelerometer data using a wearable device," in *Pattern Recognition and Image Analysis*, J. Vitrià, J. M. Sanches, and M. Hernández, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 289–296.

[7] M. Haring, D. Reinhardt, and Y. Omlor, "Pick me up and i will tell you who you are: Analyzing pick-up motions to authenticate users," *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2018.

[8] H. Wang, H. Su, K. Zheng, S. Sadiq, and X. Zhou, "An effectiveness study on trajectory similarity measures," in *Proceedings of the Twenty-Fourth Australasian Database Conference - Volume 137*, ser. ADC '13. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2013, pp. 13–22. [Online]. Available: http://dl.acm.org/citation.cfm?id=2525416.2525418

[9] E. Jovanov, V. R. Nallathimmareddygari, and J. E. Pryor, "Smartstuff: A case study of a smart water bottle," *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2016.

[10] P. Gupta and T. Dallas, "Feature selection and activity recognition system using a single triaxial accelerometer," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 6, p. 1780–1786, 2014.

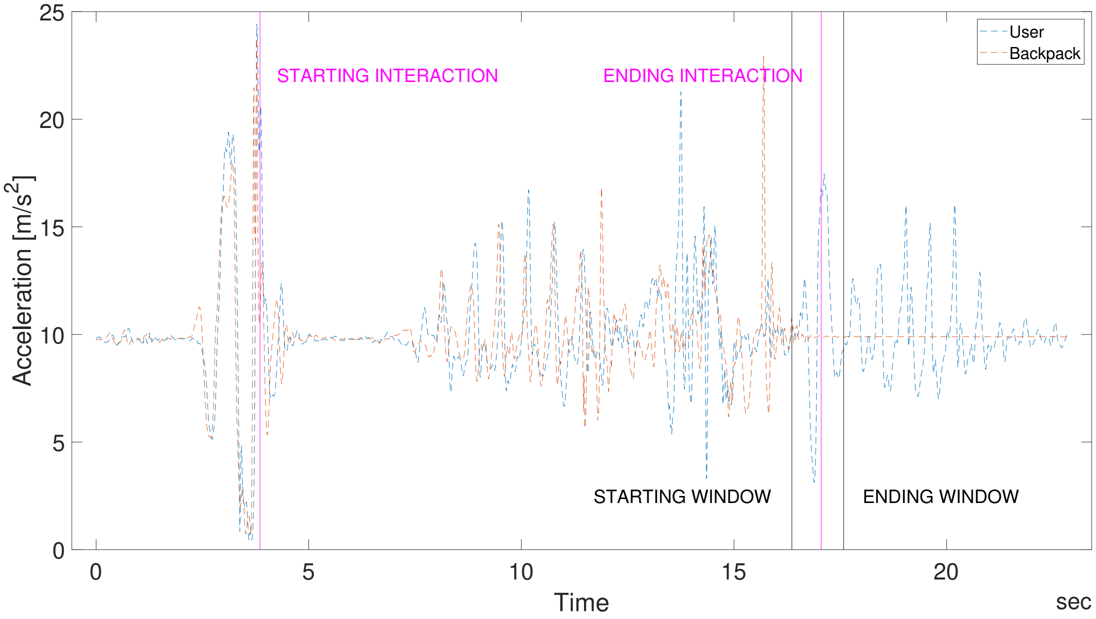

Fig. 7: Accelerometer signals of user and backpack. It's possible to see the interaction window and the misclassified window. The latter was labeled automatically as interaction, but the neural network classified it as non-interaction.