Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 11 (2017) 91 – 98

Procedia MANUFACTURING

27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy

Human behavior and hand gesture classification for smart human-robot interaction

Nuno Mendes, João Ferrer, João Vitorino, Mohammad Safeea, Pedro Neto*

University of Coimbra, Department of Mechanical Engineering POLO II, 3030-788 Coimbra, Portugal

Abstract

This paper presents an intuitive human-robot interaction (HRI) framework for gesture and human behavior recognition. It relies on a vision-based system as interaction technology to classify gestures and a 3-axis accelerometer for behavior classification (stand, walking, etc.). An intelligent system integrates static gesture recognition recurring to artificial neural networks (ANNs) and dynamic gesture recognition using hidden Markov models (HMM). Results show a recognition rate of 95% for a library of 22 gestures and 97% for a library of 6 behaviors. Experiments show a robot controlled using gestures in a HRI process.

Keywords: Human Robot Interaction; Human Behavior Recognition; Gestures; Segmentation; Accelerometer

1. Introduction

In a scenario in which robots and humans share the same environment and cooperate with each other, robotic systems need to know the human behavior to predict future actions and support decision making. Understanding human activities in the context in which they take place is challenging, since a person’s activity is driven by goals,

* Corresponding author. Tel.: +351 239 790 700; fax: +351 239 790 701.
E-mail address: pedro.neto@dem.uc.pt

doi:10.1016/j.promfg.2017.07.156

motives and needs that may be conflicting [1–4]. The HRI process is enriched with information of the human behavior, for example the robot velocity can be adjusted according to if the human is static or moving.

Intuitive HRI, for example using gestures, allows humans to focus on their own work and not in the robot programming. A robot can be instructed by a worker in the shop floor by natural means, using gestures and speech for example. In order to achieve this goal, robots should be prepared to be instructed with a high-level of abstraction from the robot language. Most of the approaches to HRI rely on a close mimicking of human-human communication using gestures, speech and the most diverse human actions. There are a number of different interaction technologies for HRI, namely data gloves [5,6], magnetic sensors [7,8], inertial sensors, Electromyography (EMG) [9], vision sensors [10–14] and hybrid solutions.

Gestures can be classified in three categories, the static gestures, the dynamic gestures and movement epenthesis. The static ones are easier to recognize because of its nature where just a frame of data (features) is required to identify them [15]. Thus, simple models of ANN have been proposed to perform an effective recognition [16]. The segmentation of static gestures is easier to be carried out than the segmentation of dynamic gestures. The dynamic gestures present some big challenges due to its spatial-temporal variability. A same gesture performed by different users can differ in shape, velocity, duration, and integrality [17]. In order to recognize dynamic gestures different approaches have been employed, for example using discrete HMM to recognize online dynamic human hand gestures [18] and using HMM for full body gesture recognition [19]. Recognitions rates above 84% were reported for a collection of seven dynamic gestures. Automatic recognition of facial emotion based on feed forward ANN and support vector regressors was presented by [20]. An interesting study in the field reports a neural-based classifier for gesture recognition with an accuracy of over 99% for a library of 6 gestures [21]. Other authors concluded that a hand contour-based neural network training is faster than complex moment-based neural network training but in the other hand the former proved to be less accurate (71%) than the latter (86%) [22]. User detection in activity and location patterns was studied by [23] who proposed a fusion-based HMM architecture.

2. Proposed approach

This paper proposes a HRI framework in which human behaviors and hand gestures are recognized and used to tele-operate a robot. The use of an accelerometer to collect representative information about human behavior is proposed. One of the goals of this study is to make use of a minimal number of sensors and at the same time being a low-cost solution to identify human behaviors that are used to adapt the way a robot performs its work. The robot can reduce its speed, stop, or move to a secure place depending on identified human behavior. Six behaviors were considered: walking, running, jumping, stand-to-sit, standing and seated.

A gesture spotting solution using an infrared stereo vision system as interaction technology, the Leap Motion Controller (LMC), is also proposed. Gesture patterns are recognized in continuous (not separately) recurring to ANNs and HMMs specifically adapted to the process of controlling an industrial robot (supervised learning). In continuous gesture recognition, communicative gestures (with an explicit meaning) appear intermittently with non-communicative gestures (transition gestures, emotional expressions, idling motion, etc.), with no specific order. In this way, it is proposed an ANN-based architecture to recognize static gestures and HMM-based architecture to recognize dynamic gestures. Experimental results demonstrated that the proposed solution presents high RRs, low training and learning time, a good capacity to generalize, it is intuitive to use and allows robot operation independently from the conditions of the surrounding environment. Fig. 1 shows a scheme of the proposed system.

2.1. Data collection

In order to collect communicative information from human to the robot, two kind of sensors are used: (1) an accelerometer to collect human behavior and an infrared vision system for collection of hand gestures. The acceleration data was collected using an Analog Devices ADXL330 tri-axial accelerometer. It measures over a range of at least +/- 3g and it yields a sensitivity accuracy of 10%. This sensor was connected to a computer over Bluetooth. Hand gestures are captured by an infrared vision system, which also converts images in hand shape features, the Leap Motion Controller (LMC).

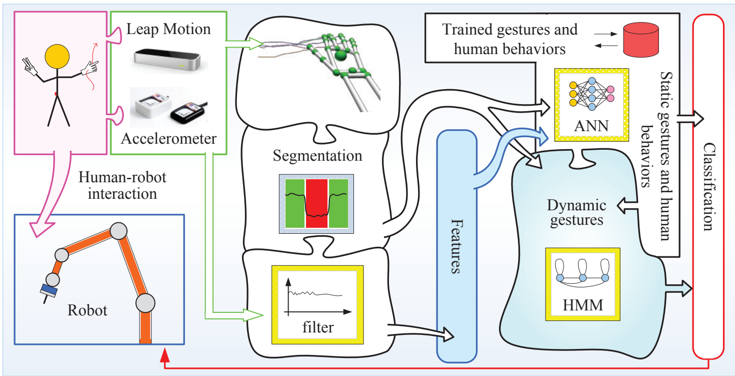

Fig. 1. Human gesture and behavior recognition architecture.

2.2. Data pre-processing

The two kind of communicative information suffer independent and different processing. The recorded data containing the accelerations along the x, y and z axes as well as the elapsed time were time-stamped and labelled with the corresponding activity name and trial number. After collecting the acceleration signals, the first step is to process these data. The “raw” signals yield noise and other unnecessary information to compute the desired features. The data is sampled at 80 Hz. Then, a high-frequency noise removal filter is applied to each signal. In this case, a median filter with a window size of 3 was used.

Each noise-filtered signal undergoes a high-pass filter to isolate the body accelerations (B_x , B_y and B_z), responsible for quick and sudden variations in the accelerations values. Subsequently, the gravity accelerations (G_x , G_y and G_z), which represent smoother oscillations in the acceleration magnitude, can be determined by subtracting the body acceleration signal from the noise-filtered one. Also, it is known that the gravity component lies in frequencies below 0.25 Hz, approximately. A Chebyshev Type II IIR high pass filter, with a cut-off frequency around 0.25 Hz, was used to separate the effect of the gravity component.

2.3. Features extraction

Several features were computed from the pre-processed acceleration signals using a window size of 128 samples with 50% overlap between consecutive windows. This overlap value demonstrated success in preliminary tests. For a sampling frequency of 80 Hz, each window corresponds to 1,6 seconds of data. A representation is given in Fig. 2.

In a first stage, two features, Mean and Mean Absolute Deviation, were extracted from each of the seven acceleration signals, $B_x, B_y, B_z, G_x, G_y, G_z$ the magnitude of the median-filtered acceleration vector, A_{abs}, giving a total of fourteen computed attributes.

A second set of features was extracted from the body accelerations. This set comprises the following features: Time-domain features: Minimum and Maximum values, Signal Magnitude Area (SMA), Tilt Angles and Correlation between axes; Frequency-domain features: Signal Energy and Signal Entropy.

In relation to hand gestures performed by a user, they are represented by features which are directly provided by the LMC. The meaning of each feature is described in Table 1. For recognition of static gestures, twenty-one input signals are provided to classifier method as input features. These inputs are directly provided by the LMC.

To recognize dynamic gestures, eighteen sets of features are provided to a recognition architecture as input signals. Each set of features consists of five features, as shown in Table 2, provided directly from the LMC. These five features were selected because they were the most influenced data in the predefined dynamic gestures. The features provided by the LMC are continuous values in the interval.

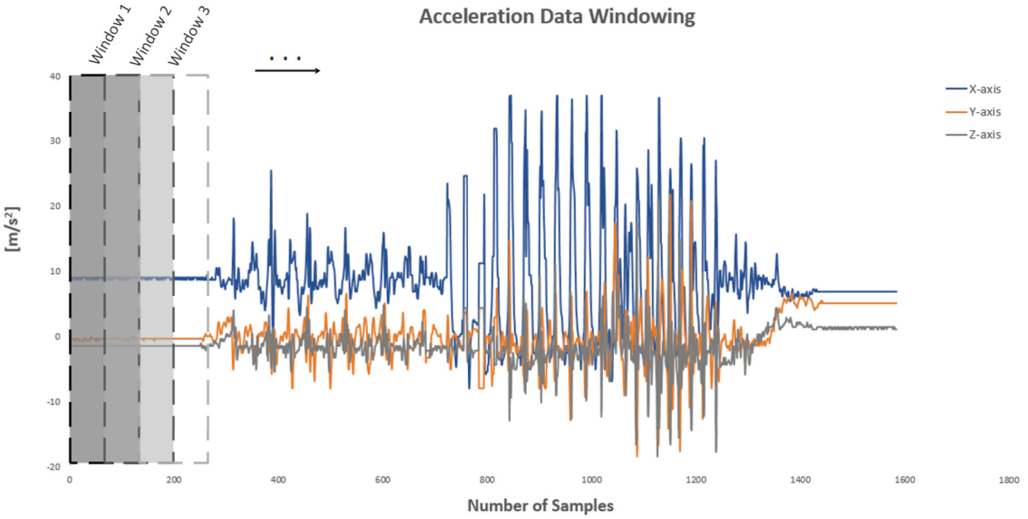

Fig. 2. Data windowing with overlapping windows.

Table 1. Features of static gestures.

Data	Description
x_1, x_2, x_3	*x*, *y* and *z* components of the normal vector to the palm hand, respectively;
x_4, x_5, x_6	*x*, *y* and *z* components of the palm hand direction vector, respectively;
x_7, x_8, x_9	*x*, *y* and *z* components of the thumb finger direction vector, respectively;
x_{10}, x_{11}, x_{12}	*x*, *y* and *z* components of the index finger direction vector, respectively;
x_{13}, x_{14}, x_{15}	*x*, *y* and *z* components of the middle finger direction vector, respectively;
x_{16}, x_{17}, x_{18}	*x*, *y* and *z* components of the ring finger direction vector, respectively;
x_{19}, x_{20}, x_{21}	*x*, *y* and *z* components of the pinkie finger direction vector, respectively.

Table 2. Features of dynamic gestures.

Data	Description
x_1, x_2, x_3	*x*, *y* and *z* components of the normal vector to the palm hand, respectively;
x_4	grab property;
x_5	*x component of the thumb finger direction vector.*

2.4. Segmentation

Gestures can be divided in two different families, static and dynamic gestures. Gesture segmentation is the task of finding the start and the end of a communicative gesture from continuous data. Since the duration of a gesture (static or dynamic) is variable this can be a challenging task [24]. For static gestures the segmentation, a static frame is considered as input for the classification. When the variability in data is considered stable, a frame of data is chosen to carry on the next stage (recognition). For dynamic gestures more than one frame of data is required to perform its recognition. In this study, the segmentation strategy consists in analyzing the variability of a threshold parameter that is established in some features received from the LMC. This threshold was established to know if there is movement of the human hand. Motion detection is done by speed analysis of two fingers, the thumb and the middle. When the speed of one of these fingers is higher than 40 mm/s, the system considers that there is a dynamic gesture being performed and proceeds to its recognition. For human behaviors, we decided not to introduce any segmentation in the process because when user changes from one behavior to other there is not a clear variation step.

2.5. Recognition

Two different classification methods are used, one based on ANN and other based on HMM. These techniques were adopted because of the nature of the data, i.e. ANN performs well for static gesture classification while HMM performs well in time series classification. These methods must present good learning capabilities with the ability to generalize and produce results from all kinds of input data: from the accelerometer sensors; and from the infrared stereo vision system, even if they are relatively different from the trained input patterns.

Two independent ANN are proposed to classify human behaviors and static/dynamic hand gestures. The proposed ANN architectures are feed-forward ones with the parameters shown in Table 3. The number of neurons in the output layer correspond to the pattern library size (number of human behaviors and number of hand gestures). The back-propagation algorithm is used as a learning/training algorithm to determine the weights of the network.

To classify human behaviors another methodology based on HMM was followed and compared with the methodology presented above. Additionally, a similar HMM methodology was also used to classify dynamic hand gestures. This latter HMM methodology is independent of the former one. A HMM was trained for each pattern, human behaviors or dynamic hand gesture. A feature vector is used to train the different HMMs and learn the model parameters. In the recognition phase, an output score is computed for each model. The model with the highest value (output score) represents the recognized gesture. However, a gesture is just accepted as recognized for the HMM with the best likelihood if the likelihood is higher than a predefined threshold.

Table 3. ANNs parameters.

	ANN for human behaviors	ANN for Hand gestures
Number of hidden layers	1	1
Number of neurons in the input layer	21	21
Number of neurons in the hidden layer	21	21
Number of neurons in the output layer	6	12

3. Experiments

To assess viability of the proposed approach, two kind of experiments were carried out. On a first experiment, each human behavior and hand gesture is recognized in an isolated test. On a second experiment, both pattern libraries are used to carried out industrial robotic tasks where an industrial robot is online controlled by the user through hand gestures. On the same time, each human behavior has a different influence on robot behavior.

In this work, we focused on six human behaviors, four dynamic and two postures: walking, running, jumping, stand-to-sit, standing, seated. These activities were performed by 2 subjects, fifty times each one and over different days. The device was worn on the waist over the right leg, while each subject was performing the activities.

Twelve static gestures (SG) and ten dynamic gestures (DG) (Fig. 3) were defined to tele-operate the robot. Two kinds of tests were performed, Test 1 and Test 2. The system was tested by five individuals that had trained the system (Test 1) and by five individuals that had not trained the system (Test 2). In both tests the users perform static and dynamic gestures. Each gesture was performed 100 times in each test.

The second experiment consists of each gesture and human behavior has a different influence in the robot. The static gestures are used to or send commands to the robot, for example: when the user performs SG1, the robot must perform a given robotic assembly task; when the user perform SG5, the robot must stop; etc. The dynamic gestures are used to online control of the robot, for example: when the user performs DG1, the robot must move itself on the positive x-axis direction; when the user performs DG7, the robot must change its end-effector orientation on the positive roll direction; etc. The human behaviors are used to allow/restrict asking some robotic function to the user, for example: if the user is walking, the robot performs its tasks in a moderate speed; if the user is running, the robot is stopped and cannot perform its tasks; if the user is seated, the robot can perform its tasks at every speed; if the user is jumping, the robot just can execute the commands associated to the static gestures; etc.

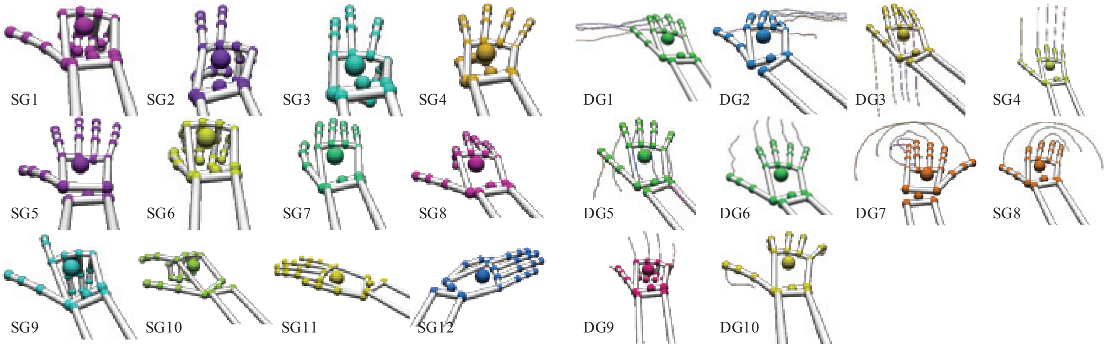

Fig. 3. Static gestures (SGs) and dynamic gestures (DGs).

4. Results and discussion

Experimental results are presented as confusion matrix. Table 4 displays results for human behavior recognition. The ANN architecture provided the best performance displaying a RR of 97.0%. Basically, two dynamic behaviors were wrongly classified: Stand-to-sit and Run. The former was classified as static Seated, which suggests that if a segmentation process was implemented, static and dynamic behaviors would be differentiated solving this point.

The HMM architecture displayed interesting results, a global RR of 65.5% is a good result since the input variables for the classifier are just raw signal without any processing. Results for the recognition of static gestures are illustrated in Table 5. By the analysis of the results it was achieved a RR of 96.1%, for test 1, which is considered relatively good compared with state of the art. This result is better than the study carried out by [22] who achieved a RR of 86.0%. An SVM classifier proposed in [25] achieved a RR of 91.0% for a library of ten static gestures. The full recognition was not achieved because the user also performs some involuntary movements that provoke some dissimilarity in gestures. Another reason for wrong recognition is because of the similarity of the gestures. Test 2 presented results close to test 1. A RR of 93.6% was obtained for gestures performed by a user that had not generated training data. This RR is lower than the RR of test 1 which was expectable because of each user performs a gesture in a slightly different way. A problem that occurs is the recognition of a gesture in the transition between two different gestures. The SG7 is frequently recognized between SG11 and SG12 and vice versa.

Results for the recognition of dynamic gestures are displayed in Table 6. These results show that from the 1000 gesture repetitions performed in test 1, the system has recognized 942 times correctly which represents a RR of 94.2%. This is a good result comparing to the study carried out by [19], which achieved a RR of 84.0% for a library of just seven gestures, and [26] that achieved a RR of just 76.0%. The hand shake introduces some wrong judgments about a DG this effect is higher with increasing distance between human hand and the center of the LMC. Test 2 presented a RR of 93.6%. The RR of test 2 is lower than the RR obtained for test 1, which was already expectable. This is because the training data may introduce some limitations in the system to recognize gestures that are performed in a different way from they have been trained.

Table 4. Confusion matrix for Human behaviors.

Recognized behavior →			ANN architecture						HMM architecture			
Performed behavior ↓	Stand	Seated	Stand-to-sit	Walk	Jump	Run	Stand	Seated	Stand-to-sit	Walk	Jump	Run
Stand	100	0	0	0	0	0	33	0	67	0	0	0
Seated	0	100	0	0	0	0	0	3	27	29	0	41
Stand-to-sit	0	9	91	0	0	0	0	0	100	0	0	0
Walk	0	0	0	100	0	0	0	0	0	100	0	0
Jump	0	0	0	0	100	0	0	0	0	0	90	10
Run	0	0	0	0	9	91	0	0	0	0	33	67

Table 5. Confusion matrix for Static Gestures.

R. →	Users had trained the system (Test 1)												Users had not trained the system (Test 2)											
P. ↓	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12
1	90	5	5	0	0	0	0	0	0	0	0	0	89	0	0	0	0	0	0	0	11	0	0	0
2	0	80	20	0	0	0	0	0	0	0	0	0	0	91	9	0	0	0	0	0	0	0	0	0
3	0	5	95	0	0	0	0	0	0	0	0	0	0	0	86	14	0	0	0	0	0	0	0	0
4	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0
5	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
6	0	0	0	0	0	95	5	0	0	0	0	0	0	0	0	0	0	97	0	0	3	0	0	0
7	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	17	0	0	83	0	0	0	0
9	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0
10	5	0	0	0	0	0	0	0	0	95	0	0	6	3	0	0	0	11	0	0	0	80	0	0
11	0	0	0	0	0	0	2	0	0	0	98	0	0	0	0	0	0	3	0	0	0	0	97	0
12	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100

Table 6. Confusion matrix for Dynamic Gestures.

Recognized gesture →	Users had trained the system (Test 1)										Users had not trained the system (Test 2)									
Performed gesture ↓	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
1	83	0	0	17	0	0	0	0	0	0	80	0	0	15	5	0	0	0	0	0
2	0	93	0	5	2	0	0	0	0	0	0	90	0	7	3	0	0	0	0	0
3	0	0	95	5	0	0	0	0	0	0	0	0	98	2	0	0	0	0	0	0
4	0	0	0	100	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0
5	0	0	0	2	98	0	0	0	0	0	0	0	0	6	94	0	0	0	0	0
6	0	0	0	9	2	89	0	0	0	0	0	0	0	4	2	94	0	0	0	0
7	0	0	1	1	0	0	98	0	0	0	0	0	1	2	1	0	96	0	0	0
8	0	0	0	4	0	0	0	96	0	0	0	0	0	5	0	0	0	95	0	0
9	0	0	0	5	3	1	0	0	90	1	0	0	0	4	3	1	0	0	92	0
10	0	0	0	0	0	0	0	0	0	100	0	0	0	2	1	0	0	0	0	97

The recognition architectures which presented the best results in the classification of each kind of pattern were used to tele-operate the industrial robot, Fig. 4, as described in section 3. Having in mind feasibility level that an industrial task needs to provide, a condition in the classification of each pattern was introduced. This condition consists of pattern validation, i.e. a given pattern is only assumed as a robot command after the pattern is continuously recognized during a time period. In the tests performed, the industrial task was successfully achieved in 100% of the cases. The delay introduced with the validation condition is perfectly negligible for the response of the robotic system.

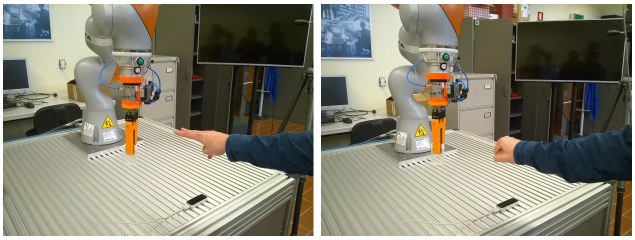

Fig. 4. Industrial robot tele-operated by gestures.

5. Conclusions and future work

A gesture and human behavior recognition system was proposed to tele-operate a robot. It demonstrated the capability to generalize from different users, even users that did not train the system before. These users can perform the same gesture trained before but with differences namely the gesture velocity, curvature of the fingers, inclination of the hand, size of the hand, etc. The proposed approach recognizes gestures in continuous and from a relatively large library of 22 different gestures (12 static gestures and 10 dynamic gestures). The system presented high performance achieving RRs of 96.1% and 94.2% for static and dynamic gestures, respectively, for the user that had trained the system. The performance of the system, when tested by a different user from that one who had trained it, was slightly lower 93.6% and 93.6% for static and dynamic gestures, respectively. The main causes for the difference in the results are attributed to the own way of each user in performing the gestures. Five human behaviors were successfully recognized by a ANN architecture, which provided RR of 97.0%. As a future work more features will be introduced in the gesture recognition architectures to improve the RRs.

Acknowledgements

This research was supported by Portugal 2020 project DM4Manufacturing POCI-01-0145-FEDER-016418 by UE/FEDER through the program COMPETE2020.

References

[1] M. Hardegger, L.-V. Nguyen-Dinh, A. Calatroni, G. Tröster, D. Roggen, in:, Proc. 2014 ACM Int. Symp. Wearable Comput. - ISWC ’14, ACM Press, New York, New York, USA, 2014, pp. 99–106.

[2] A. Moschetti, L. Fiorini, D. Esposito, P. Dario, F. Cavallo, Sensors 16 (2016) 1341.

[3] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, G. Tr, IEEE Pervasive Comput. 7 (2008) 42–50.

[4] N. Mendes, P. Neto, J. Norberto Pires, A. Loureiro, Expert Syst. Appl. 40 (2013) 1143–1151.

[5] M. Simão, P. Neto, O. Gibaru, Pattern Recognit. Lett. (2017) 1–7.

[6] P. Neto, D. Pereira, J.N. Pires, A.P. Moreira, in:, 2013 IEEE Int. Conf. Robot. Autom., IEEE, 2013, pp. 178–183.

[7] P. Neto, J.N. Pires, A.P. Moreira, in:, 18th IEEE Int. Symp. Robot Hum. Interact. Commun. 2009), IEEE, 2009, pp. 1192–1197.

[8] P. Neto, J.N. Pires, A.P. Moreira, in:, 39th Annu. Conf. IEEE Ind. Electron. Soc. (IECON 2013), IEEE, 2013, pp. 4026–4031.

[9] I. Mesa, A. Rubio, I. Tubia, J. De No, J. Diaz, Expert Syst. Appl. 41 (2014) 5190–5200.

[10] Z. Ren, J. Yuan, J. Meng, Z. Zhang, IEEE Trans. Multimed. 15 (2013) 1110–1120.

[11] P.-C. Huang, S.-K. Jeng, in:, 2012 IEEE Int. Conf. Syst. Man, Cybern., IEEE, 2012, pp. 2144–2149.

[12] A. Seal, D. Bhattacharjee, M. Nasipuri, D.K. Basu, in:, 2013 IEEE Second Int. Conf. Image Inf. Process., IEEE, 2013, pp. 597–600.

[13] R. Jafari, D. Ziou, Expert Syst. Appl. 42 (2015) 510–518.

[14] Y. Xi, S. Cho, Y.-S. Jeong, K. Cho, K. Um, in:, J.J. Park, Y. Pan, C.-S. Kim, Y. Yang (Eds.), Futur. Inf. Technol., Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 469–474.

[15] S. Gupta, J. Jaafar, W.F.W. Ahmad, Procedia Eng. 41 (2012) 827–832.

[16] C. Oz, M.C. Leu, Neurocomputing 70 (2007) 2891–2901.

[17] M.A. Simao, P. Neto, O. Gibaru, IEEE Trans. Ind. Informatics (2016).

[18] X. Wang, M. Xia, H. Cai, Y. Gao, C. Cattani, Math. Probl. Eng. 2012 (2012).

[19] F.A. Bertsch, V. V. Hafner, in:, 9th IEEE-RAS Int. Conf. Humanoid Robot., IEEE, 2009, pp. 447–453.

[20] Y. Zhang, L. Zhang, M.A. Hossain, Expert Syst. Appl. 42 (2015) 1446–1464.

[21] A.H. El-Baz, A.S. Tolba, Neural Comput. Appl. 22 (2012) 1477–1484.

[22] H. Badi, S.H. Hussein, S.A. Kareem, Neural Comput. Appl. 25 (2014) 733–741.

[23] A.R.M. Forkan, I. Khalil, Z. Tari, S. Foufou, A. Bouras, Pattern Recognit. 48 (2015) 628–641.

[24] M.A. Simao, P. Neto, O. Gibaru, in:, IECON 2016 - 42nd Annu. Conf. IEEE Ind. Electron. Soc., IEEE, 2016, pp. 809–814.

[25] G. Marin, F. Dominio, P. Zanuttigh, in:, 2014 IEEE Int. Conf. Image Process., IEEE, 2014, pp. 1565–1569.

[26] A. Kurakin, Z. Zhang, Z. Liu, in:, 20th Eur. Signal Process. Conf. (EUSIPCO 2012), 2012, pp. 1975–1979.