

# Deep Learning Enhanced Person Identification based on Smartphones Inertial Sensors

Rafael Saraiva Campos and Lisandro Lovisolo

**Abstract**—Several works explore the use of measurements from smartphones embedded inertial sensors (INS) to recognize which physical activity the person carrying the device is executing, in what is generally referred to as Activity Recognition (AR). INS readings can also be used to identify specific subjects, in what is called Activity Recognition with Person Identification (AR-PID). Here, we propose improving AR-PID capabilities by using ensemble learning with bagging (bootstrap aggregating), defining majority voting committees of deep neural networks, each of them composed by a stack of autoencoders, followed by a softmax layer. The individual inertial “signature” of a person is expected to be more distinguishable while walking than in quasi-static states (laying, sitting, standing), so we restrict our analysis to personal identification of walkers. To evaluate the proposed Deep Learning AR-PID (DL-AR-PID) scheme, we use a public domain INS database, with 10299 samples collected by 30 smartphone users. DL-AR-PID reaches accuracies above 99.6% for 29 out of 30 users, yielding an overall person identification accuracy of 98.6%.

**Keywords**—activity recognition, person identification, inertial sensors, supervised learning, bagging, deep learning, autoencoders, softmax layer

## I. INTRODUCTION

Activity Recognition (AR) is the identification of actions performed by humans using the information provided by mobile sensors [1]. Those sensors might either be stand-alone specialized ones or embedded into commercially available smartphones. AR lies within the context of Activity-Based computing, which aims at obtaining the state of the user and his environment through information provided by heterogeneous sensors [2].

The ability to identify not only the actions performed by whoever is carrying the sensors, but also who is executing them, would push the applications of Activity-Based computing one step further. This feature, referred to as Activity Recognition with Person Identification (AR-PID), was originally proposed in [3]. This capability might be useful in different areas, such as in assisted living technologies, to provide improved care to elderly and disabled persons, or in law enforcement and security, enabling authorities to expand the monitoring of suspects, missing persons, and individuals under house arrest.

Resorting to embedded sensors on-board smartphones for AR-PID has three main advantages if compared to the use of specialized stand-alone sensors:

- Smartphones have become ubiquitous in the last few years, making it unnecessary and cumbersome for the user to be forced to carry purpose-specific sensors. For instance, as far back as 2013, 91% of USA inhabitants already had a cell phone, of which just 55% were smartphones [4]. By the end of 2016, these figures had raised to 95% and 77%, respectively [5];
- Smartphones can easily report measurements through cellular and WiFi networks, while specialized stand-alone sensors would require additional communication interfaces;
- Smartphones primary use is not to monitor persons, therefore they allow uncooperative individuals (e.g., suspects being investigated and individuals under house arrest) to be unknowingly surveilled (if the proper legal warrants have been issued, of course).

This work aims at enhancing the results achieved by AR-PID with the use of DL autoencoders. To train and test the proposed DL-AR-PID feature, we use an online available database containing thousands of samples collected by different smartphone users while performing six different activities (laying, sitting, standing, walking on a horizontal surface, walking upstairs, walking downstairs) [6].

The remainder of this work is organized as follows. Section II lists and briefly discusses some works dealing with activity recognition and person identification through walking patterns. Section III describes the experimental setup and data processing carried out by the authors in [6] to build the INS database, as well as the features we selected from this original database for DL-AR-PID. Section IV briefly addresses the shallow multiclass classifier used for recognition of the aforementioned six activities, and also the shallow binary classifier for discriminating between walking and stationary states – the predictive success of which is a *sine qua non* condition for the envisioned DL-AR-PID scheme to work properly. Following, Section V delves into the details of the deep learning framework used to implement DL-AR-PID, and presents the achieved experimental results. Finally, Section VI brings a brief conclusion.

## II. RELATED WORK

### A. INS-based Activity Recognition

There are many papers on INS-based activity recognition, either using solely accelerometer data [7][8][9] or both accelerometer and gyroscope data [1][6].

In [7], the authors carried out the classification of three postures (sitting, standing, lying) from data collected by six

Rafael Saraiva Campos (CEFET/RJ), Lisandro Lovisolo (PROSAICO/PEL/DETEL/UERJ). E-mails: rafael.campos@cefet-rj.br, lisandro@uerj.br.

elderly subjects carrying a waist-mounted triaxial accelerometer. Using a Gaussian Mixture Model (GMM) based system, they achieved a mean accuracy of 91.3%.

In [8], the authors also used data from a triaxial accelerometer. The data was collected by two subjects while performing eight different activities (standing, walking, running, walking upstairs, walking downstairs, sitting, vacuuming, and brushing teeth). An accuracy of 99.57% has been reported using plurality voting with multiple classifiers ( $k$ -Nearest Neighbors, Support Vector Machines, Naive Bayes, among others). However, the small number of test subjects (only 2) reduces the significance of the results.

In [9], the authors proposed using Jerk-based feature extraction for activity recognition of cattle from acceleration data. Therefore, instead of directly using features from the triaxial accelerometer data, they employed jerk signals, i.e., changes of accelerations. A triaxial accelerometer was placed on the top of the neck collar of 21 cows, and the objective was to recognize the cows' main activities (standing, lying, walking). The authors reported an overall accuracy of 92.5% using a decision tree classifier.

In [1], the authors used both accelerometer and gyroscope data. They built an INS-measurements database and made it available online [6]. Section III-A provides more details on that, as we use their database to evaluate our DL-AR-PID scheme.

### B. INS-based Person Identification

There is much research on person identification based on walking patterns (such as gait) extracted from video images [10][11][12] or using footstep induced floor vibration [13][14]. Nonetheless, we were not able to find in the literature any previous works on INS-based person identification, except for the aforementioned work by the authors on AR-PID [3].

## III. FEATURES FOR HUMAN ACTIVITY RECOGNITION AND PERSON IDENTIFICATION

### A. INS Measurements Database

The authors in [1] compiled a human activity database and made it available online [6]. They instructed 30 individuals to carry a smartphone (Samsung Galaxy II) while performing six different activities: laying, sitting, standing, walking over a plane surface, walking upstairs and walking downstairs. The 30 individuals were then split into two disjoint sets, used in [1] as the train and test sets, respectively:

- Set  $\mathcal{A}$ : 20 individuals, who collected 7352 samples;
- Set  $\mathcal{B}$ : 10 individuals, who collected 2947 samples;

While performing the aforementioned activities, triaxial acceleration and angular velocity from the smartphone embedded INS sensors (accelerometer and gyroscope) were collected at a sampling rate of 50 Hz. These signals were then low-pass filtered for noise reduction using a 3rd order Butterworth filter with 20 Hz cutoff frequency. This was assumed to be adequate to capture body motion, as 99% of its energy is below 15 Hz [15]. Additional signals are derived from these triaxial

components, calculating their Euclidean norm and the rates of change of both linear acceleration and angular velocity. The signals were then used to build 2.56-sec blocks, with 50% overlap between adjacent blocks. Several statistics of the signals in time and frequency domain were obtained (mean, median, standard deviation, kurtosis, among others) for each block. This resulted in 561-feature vectors. Each vector is accompanied by two labels indicating the activity being performed and the person who was carrying the smartphone. A more complete description of the features can be found in [6].

### B. Selected Features

To simplify data processing and the training of the deep learning classifiers, we selected a subset of the 561 input variables used in [6]. This subset comprises just 81 variables. All these variables are calculated within each 2.56-sec block (i.e., over 128 samples, as the sampling rate is 50 Hz). Each 81-component feature vector contains [3]:

- accelerometer and gyroscope measurements for each direction ( $\mathbf{x}$ ,  $\mathbf{y}$ ,  $\mathbf{z}$ ): (1) mean, (2) standard deviation, (3) median absolute deviation, (4) maximum value, (5) minimum value, (6) energy, (7) interquartile range, (8) signal entropy and (9) inter-direction correlation ( $\mathbf{xy}$ ,  $\mathbf{xz}$ ,  $\mathbf{yz}$ ); this yields 2 sensors  $\times$  3 directions/sensor  $\times$  9 variables/direction = 54 variables;
- the Euclidean norm (considering the three components) of the first eight variables listed above, i.e.,

$$v_{\text{norm},i,j} = \sqrt{\sum_{k=1}^3 v_{i,j,k}^2} \quad (1)$$

where  $v_{i,j,k}$  is the  $k$ -th component of the  $i$ -th variable measured by the  $j$ -th sensor, with  $k = 1, 2, 3$  (corresponding to directions  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$ , respectively),  $i = 1, \dots, 8$  and  $j = 1$  (accelerometer) or 2 (gyroscope); this yields 2 sensors  $\times$  8 variables/sensor = 16 variables;

- the signal magnitude area (SMA) of the acceleration and angular velocity, considering all triaxial components, i.e.,

$$\text{SMA}_j = \frac{1}{128} \sum_{l=1}^{128} \sum_{k=1}^3 s_{l,j,k} \quad (2)$$

where  $s_{l,j,k}$  is the  $l$ -th sample of the  $j$ -th sensor in the  $k$ -th direction, with  $k = 1, 2, 3$  ( $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$ ),  $j = 1, 2$  (accelerometer and gyroscope) and  $i = 1, \dots, 128$  (the number of samples per block); this yields 2 additional variables;

- SMA of the Euclidean norm (considering the three components) of acceleration and angular velocity, i.e.,

$$\text{SMA}_{\text{norm},j} = \frac{1}{128} \sum_{l=1}^{128} s_{\text{norm},l,j} \quad (3)$$

where  $s_{\text{norm},l,j}$  is the  $l$ -th sample of the  $j$ -th sensor, taking the Euclidean norm considering components  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$ , with  $j = 1, 2$  (accelerometer and gyroscope) and  $i = 1, \dots, 128$ ; this yields 2 additional variables;

- the angles between the gravity field and the following vectors: (i) mean linear acceleration, (ii) mean linear

acceleration rate of change ( $dv/dt$ , a.k.a. jerk signal [9]), (iii) mean angular velocity, (iv) mean angular velocity rate of change ( $dw/dt$ , also a jerk signal), (v) to (vii) directions  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$ , respectively; this yields 7 additional variables.

#### IV. HUMAN ACTIVITY RECOGNITION: DISCRIMINATION BETWEEN WALKING AND STATIONARY STATES

The individual inertial “signature” of a person is expected to be more distinguishable while walking than in quasi-static states (laying, sitting, standing). Therefore, this work aims to identify specific subjects through INS-based patterns generated while walking. For such a scheme to work properly, the ability to discriminate between stationary and walking states with a very high degree of accuracy is essential, before forwarding walking measurements to the deep learning classifiers that shall recognize specific persons. So, the proposed DL-AR-PID scheme has two main stages: (i) discrimination between stationary (laying, sitting, standing) and walking states (walking over a plane surface, walking upstairs, walking downstairs); (ii) person identification using ensembles of deep learning classifiers.

For the first stage, we use the shallow binary classifier built in [3]. Sets  $\mathcal{A}$  and  $\mathcal{B}$  (see Section III-A) are used to train and test the classifier. Set  $\mathcal{A}$  provides the training and validation sets (the samples are randomly distributed as follows: 75% for training and 25% for validation), while set  $\mathcal{B}$  provides the test set. Note that the input vectors have only 81 components, as described in Section III-B. Table I summarizes the results reported in [3]. The classification accuracy reaches 99.9%, with just 2 misclassifications. This result satisfies the premise previously laid out for the viability of the proposed DL-AR-PID scheme: a very high accuracy in the separation between walking and stationary states.

TABLE I  
AR CONFUSION MATRIX (WALKING VERSUS STATIONARY STATES)

Output Class	Target Class		Precision (%)
	Stationary	Walking	
Stationary	1559	1	99.9
Walking	1	1386	99.9

#### V. DEEP LEARNING ENHANCED ACTIVITY RECOGNITION WITH PERSON IDENTIFICATION

To implement DL-AR-PID, we use ensemble learning with bagging (bootstrap aggregating) [16]. In this technique,  $M$  subsets of the training data are randomly formed by sampling with replacement, and each subset trains a different supervised classifier. The output of the  $M$  classifiers is then combined using majority voting. Outputs of majority voting committees are expected to have a lower variance than outputs of single binary classifiers, smoothing the stochastic component inherent to the deep learning neural network training and providing a more reliable and stable classification [17].

#### A. Input Vectors Repository

For AR-PID, it makes no sense employing sets  $\mathcal{A}$  and  $\mathcal{B}$  (see Section III-A) separately for training and testing, respectively, as they have samples from distinct users (20 users in set  $\mathcal{A}$  and 10 users in set  $\mathcal{B}$ ). Therefore, we take  $\mathcal{A} \cup \mathcal{B}$ , where  $\#(\mathcal{A} \cup \mathcal{B}) = 10299$  input vectors (the symbol  $\#$  indicates the cardinality of a set, i.e., its number of elements). Following, we select only the input vectors corresponding to walking states (walking horizontally, walking upstairs, and walking downstairs), which yields set  $\mathcal{C}$ , where  $\mathcal{C} \subset (\mathcal{A} \cup \mathcal{B})$  and  $\#\mathcal{C} = 4672$  input vectors distributed among 30 users, as Fig. 1 shows. Set  $\mathcal{C}$  is the repository of input vectors for training and testing the classifiers [3].

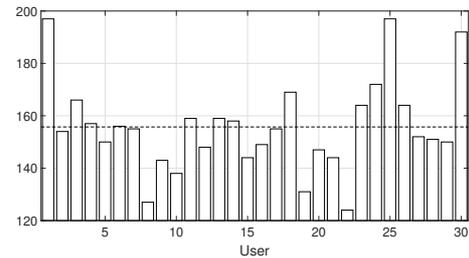


Fig. 1. Number of input vectors (walking states only) per user. The dashed horizontal line indicates the average number per user (approx. 156 input vectors)

#### B. Train and Test Sets

To compose the train set  $\mathcal{D}$ , 85% of the samples in  $\mathcal{C}$  are randomly selected. However, as Fig. 1 indicates, the number of walking samples per user in  $\mathcal{C}$  is not uniform. Then, to prevent any biasing in the DL classifiers, for each user some samples are randomly selected and replicated, so that all users contribute with the same number of input vectors in the training set  $\mathcal{D}$ . The majority voting committee has  $M = 15$  classifiers, and 15 bootstrap samples are generated, each with  $\#\mathcal{D}$  samples. So, each classifier is trained by a different set of input vectors.

#### C. Neural Networks Topology

Autoencoders (AEs), convolutional neural networks (CNNs), deep belief networks (DBNs), and recurrent neural networks (RNNs) are the most commonly used DL architectures [18]. In this work, we use sparse AEs: each of the  $M = 15$  classifiers in the majority voting committee is obtained by stacking two sparse autoencoders and a softmax layer.

Fig. 2 shows a schematic representation of an AE, with its two stages: the encoder and the decoder. The encoder can learn a more compact representation of the input data (if there are fewer neurons in the hidden layer than in the input layer), so one of its main uses is for dimensionality reduction [19]. The decoder attempts to recover the original data from the encoder output.

Each deep neural network in the majority voting committee is built by stacking two AEs. Following, a softmax layer is

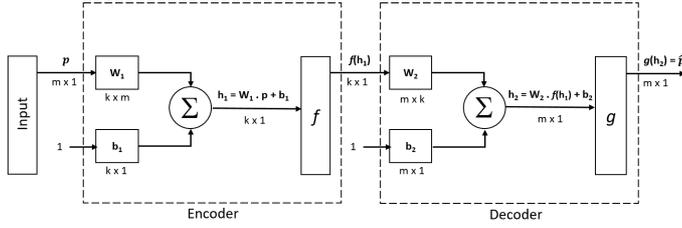


Fig. 2. Autoencoder topology, where  $m$  is the number of input nodes (i.e., the number of components of the input vector  $\mathbf{p}$ );  $k$  is the number of neurons in the hidden layer;  $\mathbf{W}_1$  and  $\mathbf{W}_2$  are the synaptic weights matrices of the encoder and decoder;  $\mathbf{b}_1$  and  $\mathbf{b}_2$  are the bias vectors of the encoder and decoder;  $f$  and  $g$  are the activation functions of the neurons in the hidden and output layers, respectively. The decoder yields an approximation  $\hat{\mathbf{p}}$  of input vector  $\mathbf{p}$ .

added. Fig. 3 depicts the deep neural network topology. For the first AE, one has  $m = 81$ , as it receives samples from set  $\mathcal{D}$  as input, and  $k = 37$ . For the second AE, one has  $m = 37$ , as it receives as inputs the outputs of the first encoder, and  $k = 15$ . The activation functions  $f$  and  $g$  of the hidden and output layers of both autoencoders are the sigmoid and linear functions, i.e.,  $f(h) = 1/(1 + \exp(-h))$  and  $g(h) = h$ . The final layer has 30 neurons (one for each class, i.e., user) with softmax transfer functions. Thus, the output  $s_i$  of the  $i$ -th neuron is given by

$$s_i = \frac{\exp(h_i)}{\sum_{j=1}^{30} \exp(h_j)} \quad (4)$$

where  $h_i$  is the aggregate input of the  $i$ -th neuron. The output class is indicated by the neuron with the highest output value in the softmax layer [16].

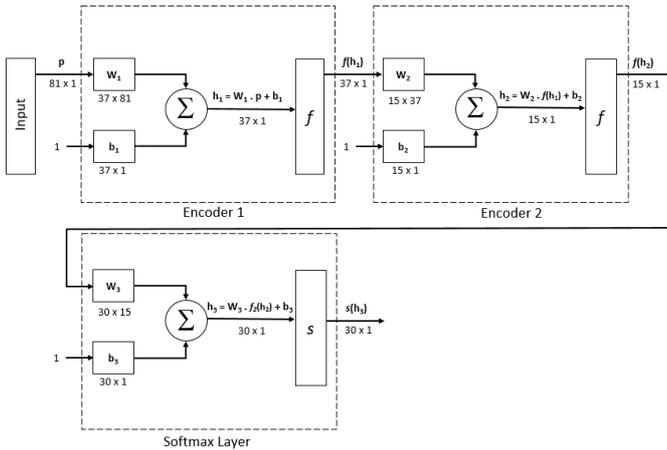


Fig. 3. Topology of the deep neural network, where  $S$  indicates a softmax layer.

#### D. Neural Networks Training

1) *AEs*: the first AE is trained using set  $\mathcal{D}$ . The second AE is trained using the outputs of the first encoder. The loss function used for batch training both AEs is the mean squared

error (MSE) with L2-norm and sparsity regularization

$$E = \frac{1}{n} \sum_{j=1}^n \sum_{i=1}^m (x_{ij} - \hat{x}_{ij})^2 + \lambda \Omega + \beta \Psi \quad (5)$$

where  $n$  is the number of vectors in the training set,  $m$  is the number of attributes per vector in the training set,  $\Omega$  is the L2-norm regularization term,  $\Psi$  is the sparsity regularization term,  $\lambda = 1$  and  $\beta = 0.001$ ,  $\mathbf{X} = [x_{ij}]_{i=1, \dots, m; j=1, \dots, n}$  is the training set, and  $\hat{\mathbf{X}} = [\hat{x}_{ij}]_{i=1, \dots, m; j=1, \dots, n}$  is the output set.

The L2-norm regularization term  $\Omega$  penalizes solutions with larger weights, which would make the AE unstable. This term is given by

$$\Omega = \sum_{j=1}^m \sum_{i=1}^k (w_{ij}^{(1)})^2 + \sum_{j=1}^k \sum_{i=1}^m (w_{ij}^{(2)})^2 \quad (6)$$

where  $\mathbf{W}_1 = [w_{ij}^{(1)}]_{i=1, \dots, k; j=1, \dots, m}$  and  $\mathbf{W}_2 = [w_{ij}^{(2)}]_{i=1, \dots, m; j=1, \dots, k}$  are the encoder and decoder weight matrices, respectively.

The sparsity regularization term  $\Psi$  is larger when the average activation value (output) of the neurons in the hidden layer (averaged over the entire training set in batch training) are distant in value from the desired average value  $\rho$ . The term  $\Psi$  is given by the Kullback-Leibler divergence [20], i.e.:

$$\Psi = \sum_{i=1}^k KL(\rho || \hat{\rho}_i) = \sum_{i=1}^k \left[ \rho \log \left( \frac{\rho}{\hat{\rho}_i} \right) + (1 - \rho) \log \left( \frac{1 - \rho}{1 - \hat{\rho}_i} \right) \right] \quad (7)$$

where  $\hat{\rho}_i$  is the average activation value of the  $i$ th neuron in the hidden layer, and  $\rho = 0.05$ . Low average activation values mean that the neurons fire just to a few samples in the training set. So, adding a sparsity regularization term to the cost function helps the encoder to learn a representation where each neuron responds to just a small subset of training examples, specializing in some features that are present only in that subset.

2) *Softmax layer*: the softmax layer is trained using the output of the second encoder; the cost function used to train it is the cross-entropy, defined by  $E = -\mathbf{T} \log(\mathbf{Y})$ , where matrices  $\mathbf{T}$  and  $\mathbf{Y}$  are the targets and the softmax layer output, respectively.

3) *Deep neural network*: finally, the network formed by stacking the aforementioned layers (as in Fig. 3) is trained using scaled conjugate gradient descent learning method and cross-entropy cost function (defined by the softmax layer). Early stopping was not used (as no part of the training set was reserved for validation), so the stop criterion is the maximum number of epochs (1000).

#### E. Results

Figs. 4 and 5 depict the classification metrics for the AR-PID [3] and DL-AR-PID schemes. The improvement achieved using deep learning is clear: Figs. 4(a,c) shows DL-AR-PID accuracies (rate of correct classifications) and specificities (rate of true negatives) close to 100% for all users, and Fig. 4(b) indicates that DL-AR-PID precision (rate of true positives) is above 95% for 28 out of 30 users. Fig 5 indicates that the DL-AR-PID accuracy, precision and specificity median values are equal to 100%.

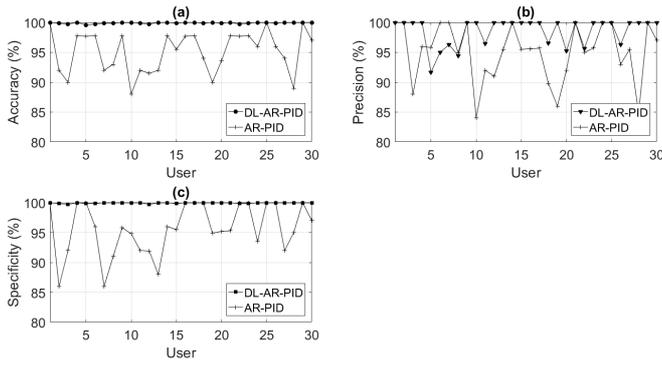


Fig. 4. Classification metrics per user: (a) accuracy, (b) precision, (c) specificity.

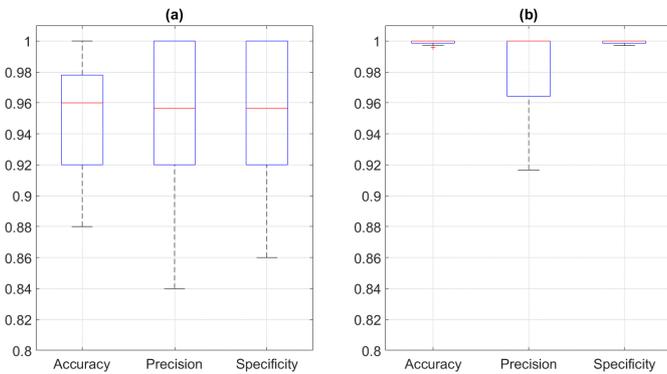


Fig. 5. Boxplot of classification performance metrics: (a) AR-PID, (b) DL-AR-PID.

## VI. CONCLUSION

In this work, we introduced enhancements to the previously proposed INS-based Activity Recognition with Person Identification (AR-PID) scheme, which resulted in a significant reduction of classification error. This improvement was implemented using deep neural networks, formed by stacking sparse autoencoders, followed by a softmax layer. The Deep Learning AR-PID (DL-AR-PID) is an extension to Activity Recognition (AR), allowing the identification of specific subjects. DL-AR-PID uses majority voting committees of deep learning classifiers. Each DL classifier is formed by stacking sparse autoencoders, followed by a softmax layer. Its performance is markedly superior in comparison to that of shallow ensemble learning. Using a public domain INS database with 10299 samples collected by 30 subjects carrying a smartphone, the proposed scheme achieved a person identification accuracy above 99.6% for 29 out of 30 users. The results show the validity and effectiveness of the proposed approach. This paves the way for DL-AR-PID being employed for security tasks, such as identification and authentication.

## REFERENCES

[1] D. Anguita, A. Ghio, L. Oneto, x. Parra and J. L. Reyes-Ortiz, “Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine”, in *Proceedings of the International*

*Workshop of Ambient Assisted Living (IWAAL 2012)*, Vitoria-Gasteiz, Spain, Dec 2012.

[2] D. Anguita, A. Ghio, L. Oneto, x. Parra, J. L. Reyes-Ortiz, “Energy Efficient Smartphone-Based Activity Recognition using Fixed-Point Arithmetic”, *Journal of Universal Computer Science - Special Issue in Ambient Assisted Living: Home Care*, Volume 19, Issue 9, May 2013.

[3] R. S. Campos and L. Lovisolo, “Person Identification based on Smartphones Inertial Sensors”, *2018 International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, Brazil, July 2018.

[4] Pew Research Center and American Life Project Spring Tracking Survey, “Cell Phone Activities 2013”, May 2013. [Online]. Available: <http://pewinternet.org/Reports/2013/Cell-Activities/Main-Findings.aspx>

[5] Pew Research Center and American Life Project Spring Tracking Survey, “Mobile Phone Ownership”, Jan 2017. [Online]. Available: <http://www.pewinternet.org/chart/mobile-phone-ownership/>

[6] D. Anguita, A. Ghio, L. Oneto, x. Parra and J. L. Reyes-Ortiz, “A Public Domain Dataset for Human Activity Recognition Using Smartphones”, in *Proceedings of the 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013)*, Bruges, Belgium, April 2013.

[7] F. Allen, E. Ambikairajah, N. H. Lovell and Branko G Celler, “Classification of a known sequence of motions and postures from accelerometry data using adapted Gaussian mixture models”, *Physiological Measurement*, Vol.27, n.10. pp.935–951, Jul 2006.

[8] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, “Activity Recognition from Accelerometer Data”, in *Proceedings of the 17th Conference on Innovative Applications of Artificial Intelligence*, Pittsburgh, USA, 2005.

[9] W. Hamalainen, M. Jarvinen, P. Martiskainen, and J. Mononen, “Jerk-based feature extraction for robust activity recognition from acceleration data”, in *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications*, Cordoba, Spain, 2011.

[10] C. BenAbdelkader, R. Cutler and L. Davis, “Stride and cadence as a biometric in automatic person identification and verification”, in *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, Washington, DC, USA, 2002.

[11] N. F. Troje, C. Westhoff and M. Lavrov, “Person identification from biological motion: Effects of structural and kinematic cues”, *Perception & Psychophysics*, Vol.67, n.4, pp. 667–675, 2005.

[12] C. Yam, M. S. Nixon, and J. N. Carter, “On the relationship of human walking and running: automatic person identification by gait”, in *Proceedings of the 16th International Conference on Pattern Recognition*, Quebec, Canada, 2002.

[13] S. Pan et al., “Indoor Person Identification Through Footstep Induced Structural Vibration”, in *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, Santa Fe, USA, 2015.

[14] D. Bales et al., “Gender Classification of Walkers via Underfloor Accelerometer Measurements”, *IEEE Internet of Things Journal*, vol.3, n.6, pp. 1259–1266, Dec 2016.

[15] D.M. Karantonis, M.R. Narayanan, M. Mathie, N.H. Lovell, and B.G. Celler, “Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring”, *IEEE Transactions on Information Technology in Biomedicine*, vol.10, pp.156–167, 2006.

[16] S. Marsland, “Machine Learning: An Algorithmic Perspective”, CRC Press, New York, 2008.

[17] F. Leisch and K. Hornik, “ARC-LH: A new adaptive resampling algorithm for improving ANN classifiers”, *Advances in Neural Information Processing Systems*, pp. 522–528, 1997.

[18] J. E. Ball, D. T. Anderson, and C. S. Chan, “Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community”, *Journal of Applied Remote Sensing*, vol.11, n.4, Sep 2017.

[19] G. Hinton and R. Salakhutdinov, “Reducing the dimensionality of data with neural networks”, *Science*, vol. 313, no. 5786, pp.504–507, 2006

[20] S. Kullback, R. A. Leibler, “On Information and Sufficiency”, *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp 79–86, 1951.