electronics

Machine Learning Techniques for Assistive Robotics

Edited by

Ester Martinez-Martin, Miguel Cazorla and
Sergio Orts-Escolano

Printed Edition of the Special Issue Published in *Electronics*

www.mdpi.com/journal/electronics

MDPI

Machine Learning Techniques for Assistive Robotics

Machine Learning Techniques for Assistive Robotics

Special Issue Editors

Ester Martinez-Martin
Miguel Cazorla
Sergio Orts-Escolano

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin

Special Issue Editors

Ester Martinez-Martin
University of Alicante
Spain

Miguel Cazorla
University of Alicante
Spain

Sergio Orts-Escolano
University of Alicante
Spain

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Electronics* (ISSN 2079-9292) (available at: https://www.mdpi.com/journal/electronics/special_issues/machine_learning_assistive_robotics).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. *Journal Name* **Year**, *Article Number*, Page Range.

ISBN 978-3-03936-338-4 (Pbk)
ISBN 978-3-03936-339-1 (PDF)

Contents

About the Special Issue Editors

Ester Martinez-Martin is Assistant Professor at University of Alicante and Subdirector at the University Institute of Computer Research. She has been a Computer Science Engineer since 2014, and she received her Ph.D. title in 2011. Her research activity has resulted in 52 publications, including a full book in Springer and 12 articles in JCR-ranked journals (6 Q1, 3 Q2, 1 Q3, 2 Q4). She has participated in 15 international and national research projects. She has also been the editor of two books in international publishing editorials and a Special Issue of a JCR Q2 journal. She has given four tutorials at international conferences (HRI, ICINCO, IAS13, IEEE RO-MAN), one guest lecture at IEEE SWYP19, and two guest lectures at prestigious centers (German Aerospace Centre (DLR) and Universidade do Minho). She has done four research stays in prestigious international centers over 18 months, financed by competitive public funds. She has participated as a member of the scientific committee for several international and national conferences, and as a member of the organizing committee seven times, three of them as "organization chair". She has also been awarded five times (MUY Young Scientist Award 2019, Award of Scientific Excellence 2013, Award of Scientific Excellence 2012, Young Researcher Award in the field of engineering and architecture 2015, Google EMEA Conference & Travel Grant 2012). She is a reviewer of scientific articles, a senior member of the IEEE, and a member of AERFAI. She has participated in various scientific dissemination activities, as well as in the promotion of STEM vocations among pre-university students, especially women. In addition, she has an extensive and successful teaching career, where she disseminates her research knowledge to the new generations.

Miguel Cazorla is a Computer Engineer from the University of Alicante (1995) and a Ph.D. in Computer Engineering from the same University (2000). In 1995, he joined the University of Alicante as a Researcher. Since 2017, he has been Full Professor. He is an IEEE Senior Member. He has completed several stays at foreign institutions (Carnegie Mellon University, University of Sydney, and University of Edinburgh). He has published more than 50 articles indexed in JCR (with more than 20 in JCR Q1) and more than 100 publications in national and international conferences. He has directed 14 Ph.D. theses and is a principal investigator in several national projects (CICYT, Challenges). He is a member of different program committees of national and international conferences. His line of research has always been centered on computational vision. From the beginning, he applied these skills to try to solve robotic tasks. Almost since its inception in research, he has worked in the processing of 3D data. In recent years, he has diversified his lines to apply deep learning techniques to different areas (medical images, object recognition, depth estimation, identification of traffic objects, etc.). All his research in recent years has been focused on social robotics and the application of all these techniques to help dependents.

Sergio Orts-Escolano received his B.Sc., M.Sc., and Ph.D. in Computer Science from the University of Alicante (Spain) in 2008, 2010, and 2014, respectively. He is currently a Research Scientist at Google. Previously, he was a Professor in the Department of Computer Science and Artificial Intelligence at the University of Alicante. Before joining academia, he worked at Microsoft Research, where he was one of the leading members of the Holoportation project (virtual 3D teleportation in real-time). He has collaborated and done multiple research/teaching stays at various top universities, including University of Edinburgh, University of Westminster, and University of Ljubljana. His research interests include computer vision, 3D sensing, robotics, real-time computing, GPU computing, VR/AR, and deep learning. He has authored 80+ publications, including 40+ in top JCR journals such as *CVIU*, *PRL*, *Neurocomputing*, and *TOG*. His work has been presented at top conferences such as CVPR, SIGGRAPH, 3DV, IROS, and BMVC. He is also a member of European networks like HiPEAC and Eucog.

electronics

Editorial

Machine Learning Techniques for Assistive Robotics

Ester Martinez-Martin [†], **Miguel Cazorla** [*,†] **and Sergio Orts-Escolano** [†]

Institute for Computing Research, University of Alicante, 03690 Alicante, Spain; ester@ua.es (E.M.-M.); sorts@ua.es (S.O.-E.)

[*] Correspondence: miguel.cazorla@ua.es; Tel.: +34-965903400

[†] These authors contributed equally to this work.

Received: 4 May 2020; Accepted: 14 May 2020; Published: 16 May 2020

1. Introduction

Assistive robots are a category of robots that share their area of work and interact with humans. Their main goal is to help, assist, and monitor humans, especially people with disabilities. To achieve this goal, it is necessary that these robots possess a series of characteristics: the ability to perceive their environment from their sensors and act consequently, to interact with people in a multimodal manner, and to navigate and make decisions autonomously. This complexity demands computationally expensive algorithms to be performed in real-time. Therefore, with the advent of high-end embedded processors, several algorithms could be processed concurrently and in real-time.

All these capabilities involve, to a greater or less extent, the use of machine learning techniques. In particular, in the last few years, new deep learning techniques have enabled a very important qualitative leap in different problems related to perception, navigation, and human understanding. In this Special Issue, various works are presented involving the use of machine learning techniques for assistive technologies, but in particular for assistive robots.

2. Machine Learning Techniques for Assistive Robotics

This Special Issue consists of eleven papers covering the application of machine learning techniques on assistive technologies and assistive robots. There are two review papers and nine research ones.

The first review [1] is focused on the identification of the research works written in English about the recognition of daily activities and environment recognition using the AdaBoost method. In particular, it focuses on the data obtained from the sensors available in mobile devices that were published between 2012 and 2018. The second one [2] reviews and summarizes the research efforts toward the development of these kinds of systems, focusing on two social groups: older adults and children with autism.

Regarding research papers, there are nine, and they are described briefly in the next paragraphs.

Pires et al. [3] use artificial neural networks (ANN) for the recognition of activities of daily living (ADLs) with the data acquired from the sensors available in mobile devices. Firstly, before ANN training, the mobile device is used for data collection. After training, mobile devices are used to apply an ANN previously trained for the ADLs' identification on a less restrictive computational platform.

In Reference [4], a system to detect the performance and the emotional state that elderly people have when performing exercises is presented. With this detection, the authors want to build an assistant that motivates those people to perform exercises and, concurrently, monitors them, observing their physical and emotional responses.

The paper presented by Ferreira et al. [5] proposes the recognition of eight ADL, e.g., walking, running, standing, going upstairs, going downstairs, driving, sleeping, and watching television, and nine environments, e.g., bar, hall, kitchen, library, street, bedroom, living room, gym, and classroom, using the instance-based k-nearest neighbor (IBk) and AdaBoost methods. The primary purpose of this paper is to find the best machine learning method for ADL and environment recognition.

The main proposal in [6] is to recognize users' environment and standing activities. Furthermore, these features are included in a framework for the ADL and environment identification. Therefore, this paper is divided into two parts: firstly, acoustic sensors are used for the collection of data towards the recognition of the environment, and secondly, the information of the recognized environment is fused with the information gathered by motion and magnetic sensors. The environment and ADL recognition are performed by pattern recognition techniques that aim for the development of their system, including data collection, processing, fusion, and classification procedures.

Modern achievements accomplished in both cognitive neuroscience and human–machine interaction technologies have enhanced the ability to control devices with the human brain by using brain–computer interface systems. In particular, the development of brain-controlled mobile robots is very important because systems of this kind can assist people, suffering from devastating neuromuscular disorders, move and thus improve their quality of life. The research work presented in [7] concerns the development of a system that performs motion control in a mobile robot in accordance with the eye blinking of a human operator via a synchronous and endogenous electroencephalography-based brain–computer interface, which uses alpha brain waveforms. The received signals are filtered in order to extract suitable features. These features are fed as inputs to a neural network, which is properly trained in order to guide the robotic vehicle.

One of the main problems in the elderly population and for people with functional disabilities is falling when they are not being supervised. Therefore, there is a need for monitoring systems with fall detection functionality. Mobile robots are a good solution for keeping the person in sight when compared to static-view sensors. Along this line, Maldonado et al. [8] propose a vision-based solution for fall detection based on a mobile-patrol robot that can correct its position in case of doubt. Deep learning-based computer vision is used for person detection, and fall classification is done by using a learning-based support vector machine (SVM) classifier.

In Reference [9], a Siamese network with an auto-encoding constraint is proposed to extract discriminative features from detection responses in a tracking-by-detection framework. The proposed network is improved to extract the previous-appearance-next vector from the tracklet for better association. Feature experiments show that the proposed Siamese network has advantages in terms of both discrimination and correctness.

Classification of complex acoustic scenes under real-time scenarios is an active domain, which has been engaged by several researchers lately from the machine learning community. In Reference [10], a framework for automatic acoustic classification for behavioral robotics is presented. Motivated by several texture classification algorithms used in computer vision, a modified feature descriptor for sound is proposed, which incorporates a combination of 1D local ternary patterns (1D-LTP) and baseline method Mel-frequency cepstral coefficients (MFCC). The extracted feature vector is later classified using a multi-class SVM, which is selected as a base classifier.

Near-infrared (NIR) facial expression recognition is resistant to illumination change. Chen et al. [11] propose a three-stream three-dimensional convolutional neural network with a squeeze-and-excitation (SE) block for NIR facial expression recognition. Each stream is fed with different local regions, namely the eyes, nose, and mouth. The experimental results on the Oulu-CASIANIR facial expression database show that the proposed method has a higher recognition rate than some of the state-of-the-art algorithms.

3. Future

The elderly population is growing year-by-year, and therefore, assistive robotics could help to improve their standard of living and quality of life. Additionally, it can also go beyond this group of people and help others that also currently live with disabilities and impairments. Machine learning techniques will help in developing robust methods in this area, creating products (robots, devices, etc.) and solutions that live together with humans on a daily basis.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial neural networks
ADL	Activities of daily living
IBk	Instance-based 32k-nearest neighbor
SVM	Support vector machine
1D-LTP	1D local ternary patterns
MFCC	Mel-frequency cepstral coefficients
NIR	Near-infrared
SE	Squeeze-and-excitation

References

1. Ferreira, J.M.; Pires, I.M.; Marques, G.; Garcia, N.M.; Zdravevski, E.; Lameski, P.; Flórez-Revuelta, F.; Spinsante, S. Identification of Daily Activites and Environments Based on the AdaBoost Method Using Mobile Device Data: A Systematic Review. *Electronics* **2020**, *9*, 192, doi:10.3390/electronics9010192.
2. Martinez-Martin, E.; Escalona, F.; Cazorla, M. Socially Assistive Robots for Older Adults and People with Autism: An Overview. *Electronics* **2020**, *9*, 367, doi:10.3390/electronics9020367.
3. Pires, I.M.; Marques, G.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C.; Zdravevski, E. Recognition of Activities of Daily Living and Environments Using Acoustic Sensors Embedded on Mobile Devices. *Electronics* **2019**, *8*, 1499, doi:10.3390/electronics8121499.
4. Costa, A.; Rincon, J.A.; Julian, V.; Novais, P.; Carrascosa, C. A Low-Cost Cognitive Assistant. *Electronics* **2020**, *9*, 310, doi:10.3390/electronics9020310.
5. Ferreira, J.M.; Pires, I.M.; Marques, G.; García, N.M.; Zdravevski, E.; Lameski, P.; Flórez-Revuelta, F.; Spinsante, S.; Xu, L. Activities of Daily Living and Environment Recognition Using Mobile Devices: A Comparative Study. *Electronics* **2020**, *9*, 180, doi:10.3390/electronics9010180.
6. Pires, I.M.; Marques, G.; Garcia, N.M.; Flórez-Revuelta, F.; Canavarro Teixeira, M.; Zdravevski, E.; Spinsante, S.; Coimbra, M. Pattern Recognition Techniques for the Identification of Activities of Daily Living Using a Mobile Device Accelerometer. *Electronics* **2020**, *9*, 509, doi:10.3390/electronics9030509.
7. Korovesis.; Kandris.; Koulouras.; Alexandridis. Robot Motion Control via an EEG-Based Brain–Computer Interface by Using Neural Networks and Alpha Brainwaves. *Electronics* **2019**, *8*, 1387, doi:10.3390/electronics8121387.
8. Maldonado-Bascón, S.; Iglesias-Iglesias, C.; Martín-Martín, P.; Lafuente-Arroyo, S. Fallen People Detection Capabilities Using Assistive Robot. *Electronics* **2019**, *8*, 915, doi:10.3390/electronics8090915.
9. Liu, P.; Li, X.; Liu, H.; Fu, Z. Online Learned Siamese Network with Auto-Encoding Constraints for Robust Multi-Object Tracking. *Electronics* **2019**, *8*, 595, doi:10.3390/electronics8060595.
10. Aziz, S.; Awais, M.; Akram, T.; Khan, U.; Alhussein, M.; Aurangzeb, K. Automatic Scene Recognition through Acoustic Classification for Behavioral Robotics. *Electronics* **2019**, *8*, 483, doi:10.3390/electronics8050483.
11. Chen, Y.; Zhang, Z.; Zhong, L.; Chen, T.; Chen, J.; Yu, Y. Three-Stream Convolutional Neural Network with Squeeze-and-Excitation Block for Near-Infrared Facial Expression Recognition. *Electronics* **2019**, *8*, 385, doi:10.3390/electronics8040385.

electronics

MDPI

Article

Pattern Recognition Techniques for the Identification of Activities of Daily Living Using a Mobile Device Accelerometer

Ivan Miguel Pires [1,2,*,†], Gonçalo Marques [2,†], Nuno M. Garcia [2,†], Francisco Flórez-Revuelta [3,†], Maria Canavarro Teixeira [4,5,†], Eftim Zdravevski [6,†], Susanna Spinsante [7,†] and Miguel Coimbra [8,†]

[1] Computer Science Department, Polytechnic Institute of Viseu, 3504-510 Viseu, Portugal
[2] Instituto de Telecomunicações, Universidade da Beira Interior, 6200-001 Covilhã, Portugal; goncalosantosmarques@gmail.com (G.M.); ngarcia@di.ubi.pt (N.M.G.)
[3] Department of Computing Technology, University of Alicante, P.O. Box 99, E-03080 Alicante, Spain; francisco.florez@ua.es
[4] UTC de Recursos Naturais e Desenvolvimento Sustentável, Polytechnique Institute of Castelo Branco, 6001-909 Castelo Branco, Portugal; ccanavarro@ipcb.pt
[5] CERNAS—Research Centre for Natural Resources, Environment and Society, Polytechnique Institute of Castelo Branco, 6001-909 Castelo Branco, Portugal
[6] Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, 1000 Skopje, Macedonia; eftim.zdravevski@finki.ukim.mk
[7] Department of Information Engineering, Università Politecnica delle Marche, 60131 Ancona, Italy; s.spinsante@staff.univpm.it
[8] Instituto de Telecomunicações, Faculdade de Ciências da Universidade do Porto, 4169-007 Porto, Portugal; mcoimbra@dcc.fc.up.pt
* Correspondence: impires@it.ubi.pt; Tel.: +351-966-379-785
† These authors contributed equally to this work.

Received: 5 February 2020; Accepted: 14 March 2020; Published: 19 March 2020

Abstract: The application of pattern recognition techniques to data collected from accelerometers available in off-the-shelf devices, such as smartphones, allows for the automatic recognition of activities of daily living (ADLs). This data can be used later to create systems that monitor the behaviors of their users. The main contribution of this paper is to use artificial neural networks (ANN) for the recognition of ADLs with the data acquired from the sensors available in mobile devices. Firstly, before ANN training, the mobile device is used for data collection. After training, mobile devices are used to apply an ANN previously trained for the ADLs' identification on a less restrictive computational platform. The motivation is to verify whether the overfitting problem can be solved using only the accelerometer data, which also requires less computational resources and reduces the energy expenditure of the mobile device when compared with the use of multiple sensors. This paper presents a method based on ANN for the recognition of a defined set of ADLs. It provides a comparative study of different implementations of ANN to choose the most appropriate method for ADLs identification. The results show the accuracy of 85.89% using deep neural networks (DNN).

Keywords: accelerometer; activities of daily living; mobile devices; sensors

1. Introduction

The accelerometer sensor commonly available in off-the-shelf mobile devices [1,2] measures the acceleration of the movement of the mobile device, enabling the recognition of activities of daily living (ADLs) [3]. After the development and conception of a system architecture for the identification of

ADLs, it could be, for example, integrated into the creation of a personal digital life coach [4], essential for the monitoring of elderly persons and persons with impairments, or for the training of certain lifestyles. The accelerometer enables the recognition of several motion activities, including running, walking on stairs, walking, and standing. Following the previous research studies [5–8], several steps are incorporated in the recognition of ADLs, including data acquisition, data processing, data cleaning, feature extraction, data fusion, and data classification.

Several authors studied the automatic recognition of ADLs [9–14]; artificial neural networks (ANN) were widely used [15,16]. The accelerometer was used for the identification of ADLs while comparing some implementations of ANN with different frameworks, such as the multilayer perception (MLP) with Neuroph [17] and Encog [18] frameworks, and the deep neural network (DNN) method with the DeepLearning4j [19] framework. The authors aimed to find the model that achieves the best accuracy in recognition of running, walking, walking downstairs, walking upstairs, and standing. These five ADLs were selected based on the literature review, wherein different studies reported reliable results for these activities, to allow the comparison with the method implemented in this research. The use of data acquired from the accelerometer sensor fused with the data retrieved from the magnetometer and gyroscope sensors is available in the literature [20]. This paper attempts to use different datasets of features with only the accelerometer data that should be analyzed to define the best combination of features. The main objective of this paper is to explore the use of different sets of features obtained using the accelerometer with the same datasets acquired for the previous study. After the comparison performed in [20] about the use of data fusion from the data acquired from the accelerometer, magnetometer, and gyroscope sensors, we verified that one of the major problems is related to the overfitting obtained during the training phase of the ANN.

The frameworks presented in this study were used in the study [20] to verify which the best methods are for the recognition of ADLs using the sensors available in the mobile device. Despite the disadvantages of achieving poor accuracy, MLP implemented with Neuroph and Encog frameworks still have the benefit of the adaption of the low resources of the mobile devices, because these methods need less power processing and memory capabilities than the DNN method implemented with DeepLearning4j. Therefore, the primary motivation of this paper is to verify whether the overfitting problem can be solved using only the accelerometer data. Additionally, the authors aim to verify the accuracy of the proposed method using only one sensor and a smaller number of features for the training of the ANN, in order to use fewer computational resources and reduce the energy expenditure of the mobile device when compared with the use of multiple sensors.

Thus, the main contribution of this paper is to perform a comparison of three different architectures of ANN methods using only the accelerometer data to verify whether the overfitting problems are avoided. This paper presents the use of ANN for ADLs recognition with the data acquired from mobile sensors. In addition, it also presents a comparative study of different implementations to find the most accurate method.

This paper is structured as follows: Section 2 presents work related to the identification of ADLs using the accelerometer sensor. Section 3 describes the steps used for the recognition of ADLs using the accelerometer sensor. Section 4 presents the discussion and results obtained during the research. Finally, Section 5 consists of the presentation of the conclusions regarding the results obtained.

2. Related Work

Several methods can be used for the automatic classification of ADLs with the data acquired from the accelerometer sensor available in the off-the-shelf mobile devices [3,21]. Numerous studies in this field are presented in the literature. Therefore, it is not possible to include them all in this document. Table 1 presents an analysis of 43 studies conducted on ADLs recognition using accelerometer data. The studies were selected according to the following criteria: (1) use of smartphones for data collection; (2) the features being clearly defined; (3) the methods being clearly defined; (4) the accuracy levels being presented. These studies are available in multiple databases such as MDPI, Springer, and ACM collected using the Google Scholar portal. Still, the vast majority have been found in the IEEE Xplore library. Following the different works analyzed, the methods that reported the best accuracies for the recognition between 1 and 8 ADLs are the different types of ANN, including MLP and DNN methods, using statistical features.

The studies presented in Table 1 reported that the most recognized ADLs with reported average accuracies high than 85% are walking, standing, walking upstairs, walking downstairs, and running. Therefore, these activities are considered in the proposed method. In total, 31 studies use smartphones located in the user's pocket. However, some studies also located the smartphone around the waist, forearm, and wrist. Moreover, some studies combine the use of smartphones with other wearable sensors.

The ADLs recognition indicates an average accuracy between 87.93% and 88.80% using different methods. In addition, the ADLs reporting better accuracies in the analyzed studies are walking, standing, walking upstairs, walking downstairs, and running. In total, 91% (N = 39) of the analyzed papers support walking recognition reporting an average accuracy of 88.80%. The standing activity is included in 29 studies which represent 67% of our literature review and provide an average accuracy of 88.65%. Walking upstairs and downstairs activities are supported by 25 (58%) and 23 (53%) studies, respectively. The first reports an average accuracy of 85.88% and the second reports an average accuracy of 85.5%. Finally, the running activity is assessed by 42% (N = 18) of the evaluated studies and reports 87.93% average accuracy.

Regarding the ADLs recognized in the analyzed studies, the mean, standard deviation, maximum, minimum, correlation, variance, and median are the most used features in the literature. In total, 86% (N = 37) of the analyzed papers use the mean feature, reporting an average accuracy of 85.74%. The standard deviation feature is included in 30, representing 70% of the evaluated papers, and provides an average accuracy of 86.70%. The maximum and minimum values are included in 19 (44%) and 17 (40%) studies, respectively. The maximum feature reports an average accuracy of 87.47%, and the minimum feature reports 88.50%. The median and correlation features are used in 10 studies (23%) each and report average accuracies of 87.44 % and 91.52%, respectively. Eight studies include the variance as a feature for ADLs recognition reporting and average accuracy of 90.15%.

The implementations that reported an accuracy higher than 88% are ANN, multi-column bidirectional long short-term memory (MBLSTM), Bayesian network, and random forest methods, reporting an average accuracy between 88.65% and 91.29%. In total, 40% (N = 17) of the analyzed papers use ANN methods reporting the average accuracy of 91.29%. Eight studies propose the random forest for ADLs recognition, reporting 90.53% average accuracy. The MBLSTM method provides 89,4% average accuracy, and the Bayesian Network is used by three studies reporting an average accuracy of 88.65%.

In summary, the number of ADLs recognized with the different methods used, as well as the particular dataset, influenced the accuracies reported. The identification of a lesser amount of ADLs reported the best results in the literature. Following the ADLs and methods that reported the best results, our research is focused on the implementation of ANN for the recognition of five ADLs, including standing, walking, running, and walking upstairs and downstairs. These ADLs were selected for our implementation because they are the most recognized in the literature, reporting reliable accuracies.

Table 1. Summary of the studies available in the literature.

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[22]	8	Standing; Sitting; Laying; Walking; Walking upstairs; Walking downstairs; Running; Nordic walking	Standard deviation; mean; maximum; minimum	73% (Majority Vote Naïve Bayes Nearest Neighbor algorithm (MVNBNN))	Smartphone located in trouser front pocket
[23]	3	Walking; running; walking upstairs	Mean; standard deviation; euclidean norm of mean; euclidean norm of the standard deviation; correlation values; 25th and 75th percentile values; frequency; amplitude; peak frequency; number of peak values	95% (KNN); 89% (Random Forest); 99% (SVM)	Smartphone in a pouch and located around waist
[24]	3	Slow walk; brisk walk; sitting	Mean; standard deviation; variance	90.9% (SVM)	Smartphone located in trouser front pocket
[25]	6	Standing; Sitting; Lying; Walking Upstairs; Walking Downstairs; Walking	Minimum; Maximum; Mean; Standard Deviation; SMA; Signal Vector Magnitude; Tilt Angle; Power Spectral Density (PSD); Signal Entropy; Spectral Energy	93.52% (Decision Tree); 69.72% (SVM); 87.2% (MLP)	Smartphone located in trouser pocket freely chosen by the user
[26]	4	walking downstairs; walking upstairs; walking; jogging	Mean; Variance; Standard Deviation; Maximum; Minimum; Correlation Coefficient; Mean Crossing Value; Peak; Spectral Energy; Power Spectral Density; Interquartile Range; DT-CWT	68.56% (SVM); 90.35% (Random Forest); 94.65% (MLP); 85.99% (J48 Decision Tree); 93.44% (KNN); 80.32% (Naive Bayes)	Smartphone located into the right jeans pocket
[27]	6	Sitting; standing; laying; walking; walking upstairs; walking downstairs	Mean; Standard deviation; Median absolute deviation; Maximum; Minimum; Signal magnitude area; Sum of the squares separated by the quantity of values; Interquartile range; Entropy; Autoregression coefficients; correlation coefficient; index of the frequency segment with biggest magnitude; Weighted average of the frequency segments to acquire a mean recurrence; skewness; kurtosis; Energy of a recurrence interval inside the 64 containers of the FFT of every window; Angle between two vectors	97.77% (Decision Tree); 89.99% (KNN); 95.55% (Naive Bayes); 100% (Random Forest); 95.55% (SVM)	Smartphone located on the waist

Table 1. *Cont.*

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[28]	1	Walking	Maximum; Minimum; Mean; Range; RMS; Standard Deviation; Zero Crossing Rate; Kurtosis; Spectral Slope	97.80% (SVM); 97.64% (Random Forest); 97.64% (Logistic); 98.11% (MLP)	Smartphones located into users' pocket freely chosen by them
[29]	1	falling	average absolute acceleration variation; impact duration; maximum; peak duration; activity level of a window that contains the impact; average acceleration of free-fall stage; number of steps; skewness; kurtosis; interquartile range; power of the impact; standard variation of the impact; square of the highest coefficient; number of peaks	97.53% (KNN)	Not available
[30]	5	jogging; walking; sitting; laying down; standing	Mean; maximum; minimum; median; SMA; Median deviation; PCA; interquartile range	94.32% (SVM); 98.74% (MLP); 91.10% (Naive Bayes); 99% (KNN); 98.80% (Decision Tree); 99.01% (kStar)	Smartphones located into users' pocket freely chosen by them
[31]	6	walking; standing; travel by car; travel by bus; travel by train; travel by metro	Mean; Median; Maximum; Minimum; RMS; standard deviation; interquartile range; minimum average; maximum average; maximum peak height; average peak height; entropy; FFT spectral energy; Skewness; kurtosis	95.6% (J48 Decision Tree); 92.4% (SMO); 61.9% (Naïve Bayes)	Smartphone in the pocket (not specified)
[32]	1	playing tennis	Mean; Variance; correlation	98.12% (Naïve Bayes); 99.61% (MLP); 99.91% (J48 Decision Tree); 100% (SVM)	Smartphone located on forearm and in the subject front pocket
[33]	1	playing fosball	Mean; Variance; Covariance; Energy; entropy	95% (MLP)	Smartphone located on pocket and smartwatch located on wrist
[34]	7	walking; jogging; walking upstairs; walking downstairs; standing; sitting; lying down	mean and standard deviation for each axis; bin distribution; heuristic measure of wave periodicity	90% (Random Forest)	Smartphone located in front pants pocket

Table 1. *Cont.*

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[35]	5	walking; standing; running; walking upstairs; walking downstairs	Mean; Variance; quartiles	80% (Sliding-Window-based Hidden Markov Model (SW-HMM))	Smartphones located on belt, right jeans pocket, right arm, and right wrist
[36]	5	running; walking; sitting; walking upstairs; walking downstairs	Mean; Variance; standard deviation; median; maximum; minimum; RMS; zero crossing rate; skewness; kurtosis; spectral entropy	80% (SVM)	4 smartphones located in the left upper arm, the shirt-pocket, the jeans front pocket, and the behind jeans pocket
[37]	6	walking; walking upstairs; walking downstairs; sitting; standing; laying	Mean; standard deviation	83.55% (Hidden Markov Model Ensemble (HMME))	Smartphone located on the waist
[38]	4	walking; running; standing; sitting	Mean; Maximum; Minimum; Median; standard deviation	99% (MLP)	Smartphone located in the user's pants pocket
[13]	4	walking; running; standing; sitting	Mean; Minimum; Maximum; standard deviation	92% (Clustered KNN)	Smartphone located in the user's jeans pocket
[39]	4	walking; running; sitting; standing	Mean; Variance; bin distribution in time and frequency domain; FFT spectral energy; correlation of the magnitude	98.69% (Decision Tree)	Smartphone located in the user's trousers pocket
[40]	5	standing; walking; walking upstairs; walking downstairs; running	Mean; standard deviation; percentiles	92% (MLP)	Smartphone located at four locations: two front trousers pockets and two back trousers pockets
[41]	6	standing; sitting; walking upstairs; walking downstairs; walking; jogging	Dual-tree complex wavelet transform (DT-CWT) statistical information and orientation	76% (Random Forest); 73.8% (Instance-based learning (IBk)); 67.4% (J48 Decision Tree); 67.4% (J-Rip)	Smartphone located in the user's trousers pocket
[42]	6	walking downstairs; jogging; sitting; standing; walking upstairs; walking	Minimum; Maximum; Mean; standard deviation; zero crossing rate for each axis; correlation between axis	92.4% (J48 Decision Tree); 91.7% (MLP); 84.3% (Likelihood Ratio (LR))	Smartphone located in their front trousers leg pocket
[43]	7	walking; running; standing; sitting; lying; walking upstairs; walking downstairs	Mean; Minimum; Maximum; standard deviation	77% (DNN)	Smarphone located in the right pant pocket

Table 1. *Cont.*

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[44]	5	running; walking; standing; sitting; laying	Mean; Median; Maximum; Minimum; Root Mean Square (RMS); standard deviation; interquartile range; energy; entropy; skewness; kurtosis	99.5% (Decision Tree)	Smartphone located in the belt or in the trousers front pocket
[45]	4	walking; running; cycling; hopping	RMS; Variance; Correlation; energy	97.69% (SVM)	Smartphone located in the pants front pocket
[46]	3	walking upstairs; walking up on an escalator; walking on a ramp	mean, standard deviation, skewness, kurtosis, average absolute deviation, and pairwise correlation of the tree axis of accelerometer; mean of the resultant acceleration	80.59% (Decision Tables); 82.97% (J48 Decision Tree); 87.49% (Naïve Bayes); 89.20% (KNN); 87.86% (MLP)	Smartphone located in the right or left palms in front of the body
[47]	4	walking; cycling; running; standing	Mean; standard deviation; correlation; power spectral density	98% (Naïve Bayes); 83% (KNN); 95% (Decision Tree); 96% (SVM)	Smartphone located along the waist in the front pocket
[48]	5	standing; walking; running; walking upstairs; walking downstairs	Mean; Median; Variance; standard deviation; maximum; minimum; range; RMS; FFT coefficients; FFT spectral energy	88.32% (Decision Tree)	Smartphone located in different positions such as in the bag, trouser pocket and hands.
[49]	5	walking; sitting; standing; walking upstairs; walking downstairs	Mean; standard deviation; variance	92.44% (KNN); 90.77% (Decision Tree); 90.4% (rule-based learner (JRip)); 92.91% (MLP)	Smartphone located in the user's trouser pocket
[50]	6	walking; jogging; walking upstairs; walking downstairs; sitting; standing	energy and variances of the coefficients of discrete wavelet transform (DWT)	79.9% (Naïve Bayes); 82.3% (MLP)	Smartphone located on the upper crevice of a user's back
[51]	3	walking; jogging; running	number of peaks; number of troughs; difference between the maximum peak and the minimum trough; sum of all peaks and troughs	93.4% (J48 Decision Tree + Decision Table + Naïve Bayes)	Smartphone positioned on the palm, front trouser pocket, backpack, and top jacket pocket
[52]	1	walking	Mean; standard deviation	98% (MLP)	Smarphone located in the user's pocket

Table 1. *Cont.*

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[53]	6	walking; jogging; walking upstairs; walking downstairs; sitting; standing	Mean; standard deviation; average absolute difference; average resultant acceleration; time between peaks; binned distribution	85.1% (J48 Decision Tree); 78.1% (logistic regression); 91.7% (MLP); 37.2% (Straw Man)	Smartphone located in the user's front pants leg pocket
[54]	5	walking; standing; sitting; walking upstairs; walking downstairs	mean, standard deviation and correlation of the raw data; energy of FFT; mean and standard deviation of the FFT components in the frequency domain	95.62% (Bayesian Network); 97.81% (Naïve Bayes); 99.27% (KNN); 93.53% (JRip)	Smartphone located in the user's right trouser pocket
[55]	5	walking; sitting; standing; walking upstairs; walking downstairs	Mean; standard deviation; variance; FFT energy; FFT information entropy	91.37% (Decision Tree); 94.29% (KNN); 84.42% (SMO)	Smartphone located in the user's trouser pocket
[56]	6	travel by car; travel by bus; travel by train; walking; travel by bike; standing	average speed; average acceleration; average bus closeness; average rail closeness; average candidate bus closeness	91.6% (Naïve Bayes); 92.5% (Bayesian Network); 92.2% (Decision Trees); 93.7% (Random Forest); 83.3% (MLP)	Smartphone located in the user's waist, arm, pocket, or bag
[57]	11	sleeping; eating; personal care; working; studying; household work; socializing; sports; hobbies; mass media; travel by car	average of acceleration; Mean Absolute Difference (MAD) of the acceleration	20.76% (SVM)	Smartphone located in the user's arm
[58]	11	walking; reading; lying down; standing; rearranging books; picking up golf or tennis balls; cycling; falling down; eating; washing hands	minimum; maximum; average; median; standard deviation; toughs and peaks of acceleration	72% (Hybrid model)	Smartphone located in the user's arm
[59]	5	walking; jogging; walking upstairs; walking downstairs; standing	mean value; mean absolute value; difference between maximum and minimum value; total value of absolute differences	96% (k-NN)	Smartphone located in the user's waist

Table 1. *Cont.*

Study	Number of ADLs	ADLs Recognized	Features	Proposed Methods and Accuracy	Device Location
[60]	6	standing; walking; walking upstairs; walking downstairs; running; hopping	FFT; 42-dimensional time domain features	72.62% (Autoregressive (AR) Model)	Smartphone located in different locations: Pants' front pocket (left), Pants' front pocket (right), Pants' back pocket (left), Pants' back pocket (right) and Jacket's inner pocket
[61]	7	running; walking upstairs, walking downstairs; walking; standing; lying down	average; median; Standard deviation	90.2% (IBk); 88.2% (Random Florest); 85.5% (Random Tree); 88.1% (J48); 80.3% (JRip); 85.8% (RepTree); 82.9% (MLP)	Smartphone located the user's leg and waist and wearable sensor located in the chest
[62]	6	running; walking; standing; walking upstairs; walking downstairs	standard deviation; mean; percentiles	90.85% (Naïve Bayes); 87.35% (K-NN); 81.16% (SVM)	Smartphone is located in the front-right and the back-left pockets
[63]	5	jumping; running; walking; walking downstairs; walking upstairs	average acceleration; peaks	83.8% (SVM); 83.4% (Empirical risk minimization (ERM)); 79.4% (K-NN); 86.8% (Bidirectional Long Short-Term Memory (BLSTM)); 89.4% (Multi-column Bidirectional Long Short-Term Memory (MBLSTM))	Not available

3. Methods

Based on the literature combined with the proposed system architecture for the recognition of ADLs in [5–7,64], the methods that should be defined for each module of the proposed system, are as follows: data acquisition, data processing, data fusion, and data classification. The data processing methods include data cleaning and feature extraction methods. Additionally, since this study only uses a single sensor, i.e., the accelerometer, the data fusion methods are not necessary.

Figure 1 represents the methodology and system architecture proposed by the authors in this paper. The data acquisition is performed using the accelerometer sensor available in commonly used, off-the-shelf mobile devices with a mobile application during running, walking, standing, and walking upstairs and walking downstairs activities. This acquired data is processed using data cleaning and feature extraction methods. After data processing, MLP and DNN methods are used for ADLs identification.

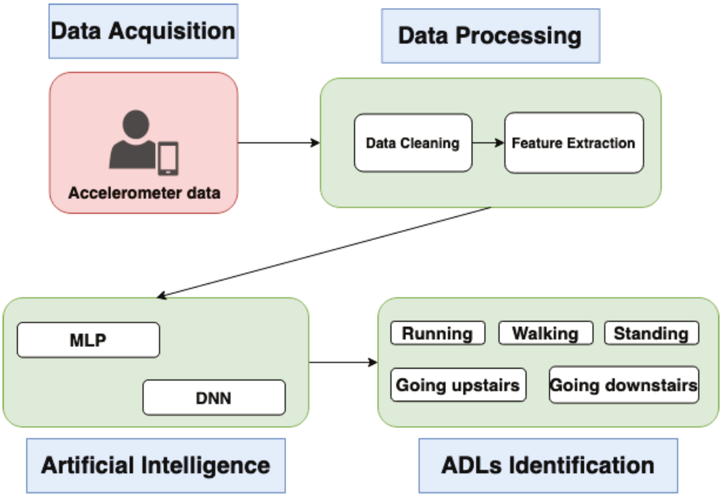


Figure 1. Methodology and system architecture for the recognition of activities of daily living (ADLs).

3.1. Data Acquisition

This study was based on the data previously acquired for the study [20], which consists on the acquisition of data related to five ADLs, such as standing (Figure 2), walking (Figure 3), running (Figure 4), walking upstairs (Figure 5), and walking downstairs (Figure 6). The data used for this study are available in a public repository [65] previously used in [20]. A visual presentation of the data collected in each activity is presented in Figures 2–6.

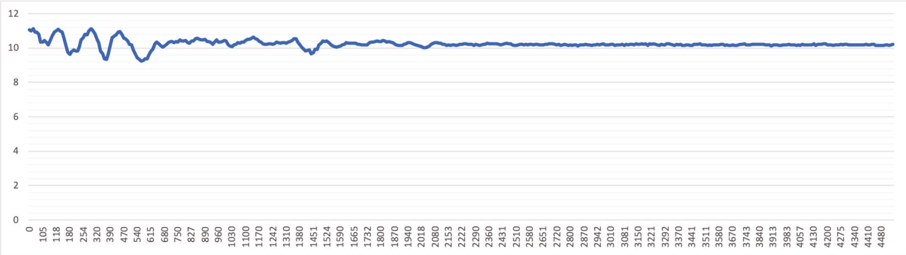


Figure 2. Acceleration (m/s^2)—five seconds of data collected during the activity of standing.

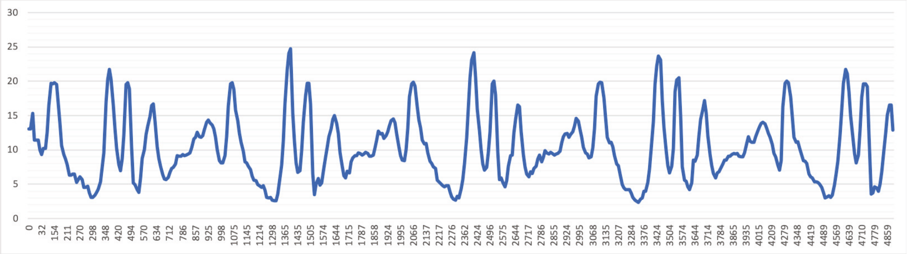


Figure 3. Acceleration (m/s^2)—five seconds of data collected during the activity of walking.

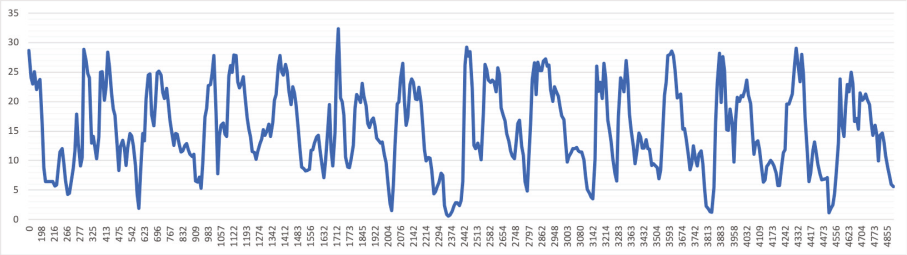


Figure 4. Acceleration (m/s^2)—five seconds of data collected during the activity of running.

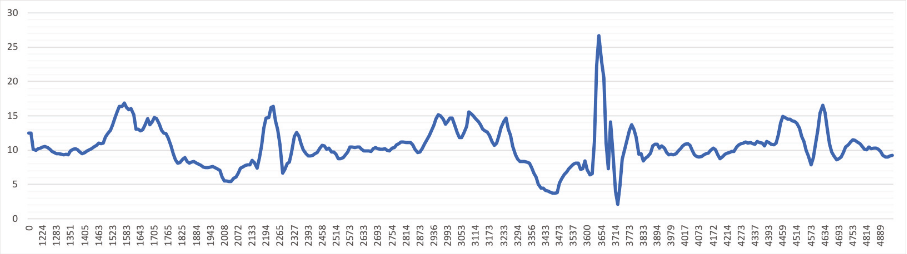


Figure 5. Acceleration (m/s^2)—five seconds of data collected during the activity of walking upstairs.

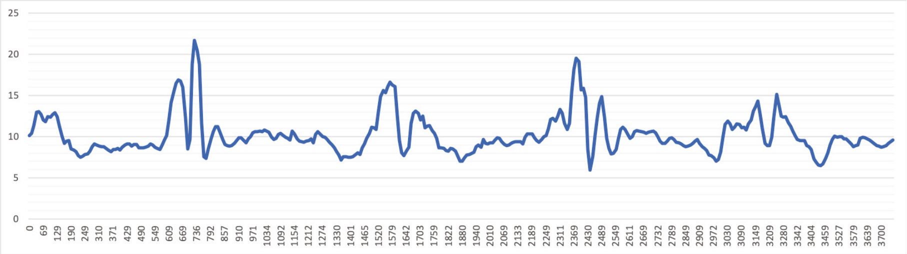


Figure 6. Acceleration (m/s^2)—five seconds of data collected during the activity of walking downstairs.

The dataset comprehends more than 2000 samples with five seconds of accelerometer data for each ADL. A mobile device placed on the front pocket of the user's pants was used for data acquisition. The data were acquired in a controlled environment, where, before the start of the data collection, the user had to select the ADLs that he/she would perform. Every five seconds of data were acquired every five minutes. When the user planed to perform another ADLs, he/she should stop the data collection and change the ADLs selected in the mobile application used.

Twenty-five individuals were selected for the experiments that always used the same mobile device; i.e., an *BQ Aquaris 5.7* smartphone [66]. These individuals were aged between 16 and 60 years old, composed of five teenagers and five people between 40 and 60 years old, and the remaining were randomly selected. Several environmental constraints were uncontrolled during the data acquisition, but we had control of the procedures related to the labeling of the different samples and the positioning of the device. As we acquired five seconds of data every five minutes, the individuals spent around 7 h performing each ADL collected by the mobile device. In total, each individual spent around 35 h for the data acquisition.

3.2. Data Processing

This study comprehends the use of accelerometer data with a low-pass filter application to clean the data [67,68]. It consists of the first step of the data processing, and this module is finalized with the extraction of the different statistical features. They are the same as the ones described in [20], but only provided by the accelerometer data, including the five largest distances between the maximum peaks; the mean, standard deviation, variance, and median of the maximum peaks; and the standard deviation, mean, maximum and minimum values, variance, and median of the raw signal.

3.3. Data Classification

For the same purpose as [20], but only with the accelerometer data, this study aimed to recognize the five proposed ADLs being used, based on the datasets presented in Figure 7. The granularity of the features included varies between the datasets 1–5; i.e., the dataset 5 contains all inputs of datasets 1 to 5.

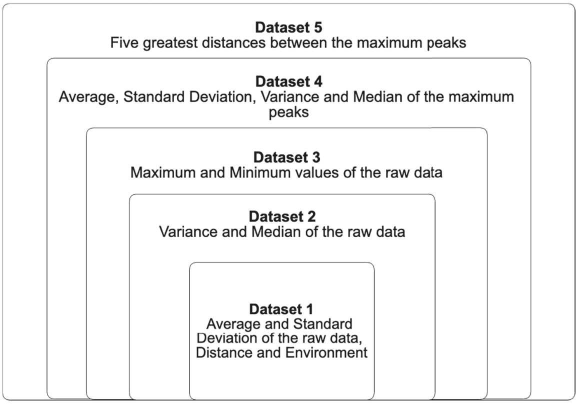

Dataset 5
Five greatest distances between the maximum peaks
Dataset 4
Average, Standard Deviation, Variance and Median of the maximum peaks
Dataset 3
Maximum and Minimum values of the raw data
Dataset 2
Variance and Median of the raw data
Dataset 1
Average and Standard Deviation of the raw data, Distance and Environment

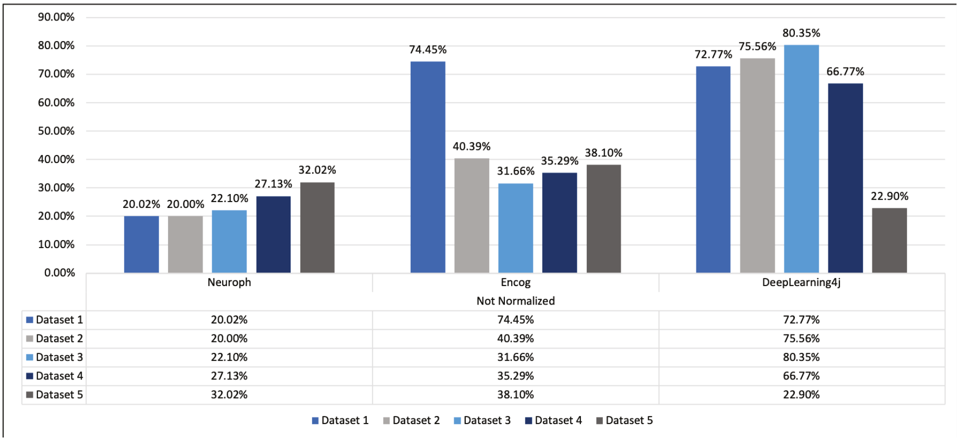
Figure 7. Datasets created for the analysis and recognition of the different ADLs.

For this purpose, we used three different implementations with distinct configurations using free software available online. The application of the MLP method takes into account the same settings, but two different implementations were performed using the Neuroph [17] and Encog [18] frameworks. Additionally, we used the DeepLearning4j framework for the application of a DNN method [19]. These are Java-based frameworks that allow for the implementation of machine learning methods with the adaptation to our data. All configurations of the frameworks implemented the sigmoid function as the activation function, a maximum of 4×10^6 iterations and backpropagation [69]. However, the learning rates applied in the MLP implementations and the DNN method are different; the value was 0.6 for MLP implementations and 0.1 for the DNN method. The MLP implementations also included the momentum value equal to 0.4. Regarding the numbers of hidden layers, the MLP methods did not include hidden layers, but the DNN method implemented three hidden layers. The DNN method also included the Xavier function [70] as a weight/initialisation function, a seed value equal to 6, and L_2 regularization [71]. After different tests and adjustments, we verified that these parameters reported more consistent results with the data acquired than others, suggesting its implementation in the developed method.

Additionally, the data classification was tested with normalized and non-normalized data, implemented the min-max normalization for the implementations of the MLP method, and the normalization with mean and standard deviation for the implementation of the DNN method.

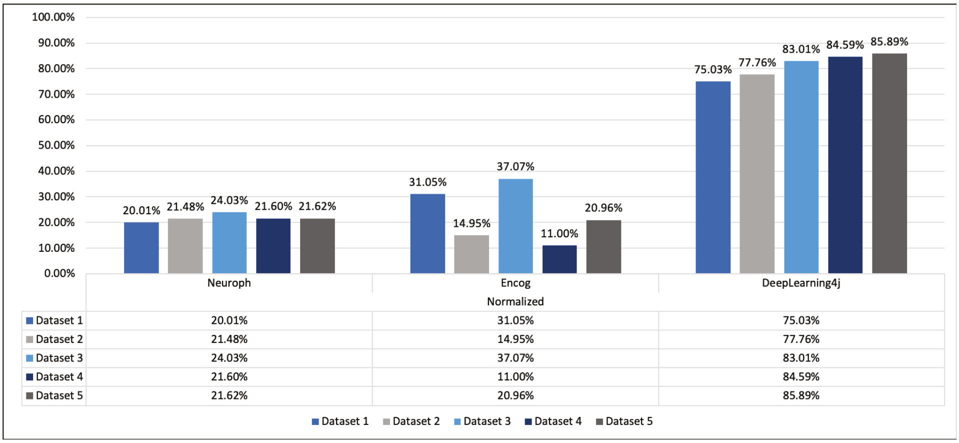
4. Results and Discussion

As the different implementations reported the existence of overfitting during the creation of the different ANNs, the early-stop training technique was implemented, stopping the training at a limit of 4×10^6 iterations. Thus, the results reported are presented in Figures 8 and 9 for non-normalized and normalized data, respectively.



Not Normalized	Neuroph	Encog	DeepLearning4j
■ Dataset 1	20.02%	74.45%	72.77%
■ Dataset 2	20.00%	40.39%	75.56%
■ Dataset 3	22.10%	31.66%	80.35%
■ Dataset 4	27.13%	35.29%	66.77%
■ Dataset 5	32.02%	38.10%	22.90%

Figure 8. Results obtained with the MLP method implemented using non-normalized data with Neuroph and Encog frameworks, and the DNN method implemented with the DeepLearning4j framework (horizontal axis) for the different datasets (series), obtaining the accuracies in percentages (vertical axis).



Normalized	Neuroph	Encog	DeepLearning4j
■ Dataset 1	20.01%	31.05%	75.03%
■ Dataset 2	21.48%	14.95%	77.76%
■ Dataset 3	24.03%	37.07%	83.01%
■ Dataset 4	21.60%	11.00%	84.59%
■ Dataset 5	21.62%	20.96%	85.89%

Figure 9. Results obtained with the MLP method implemented using normalized data with Neuroph and Encog frameworks, and the DNN method implemented with the DeepLearning4j framework (horizontal axis) for the different datasets (series), obtaining the accuracies in percentages (vertical axis).

After the implementation with the Neuroph framework, the results obtained had very low accuracies with normalized (between 20% and 30%) and non-normalized (between 20% and 40%) data. Following the implementation with the Encog framework, the results obtained had a very low accuracy (between 20% and 40%) with data without normalization, wherein, as excepted, the neural networks trained with the dataset 5 reported a certainty around 75%. When the data were normalized, the accuracy of the implemented method was always between 10% and 40%.

Next, for the implementation with the DeepLearning4j framework, the results obtained are higher than 70%, but, for data without normalization, the results reported with the dataset 5 have an accuracy lower than 30%, and for the normalized data, the results decrease with a reduced number of features—dataset 5 reported the best results.

There are two types of normalization implemented with the data acquired, including the one based on mean and standard deviation and the other one based on min-max. The accuracy reported for non-normalized data is better than the accuracy reported for data with min-max normalization. However, the results with all defined datasets increase with the application of L_2 regularization and normalization with mean and standard deviation.

Table 2 shows the maximum accuracies obtained with the MLP method with Neuroph and Encog frameworks and the DNN method with the DeepLearning4j framework. The DeepLearning4j framework reported the best accuracy, and the results obtained by Neuroph and Encog frameworks are not satisfactory.

Table 2. Best accuracies obtained with the different frameworks and datasets.

	Type of ANN	**Framework**	**Dataset**	**Best Accuracy Achieved (%)**
Non-normalised data	MLP	Neuroph	5	32.02
		Encog	1	74.45
	DNN	DeepLearning4j	5	80.35
Normalised data	MLP	Neuroph	3	24.03
		Encog	2	37.07
	DNN	DeepLearning4j	5	85.89

Analyzing the results presented in Table 2, Neuroph framework always reported bad results with an accuracy of 32.02% using dataset 5 using non-normalized data, and an accuracy of 24.03% with dataset 3 using normalized data. Among the frameworks used in this study, the Neuroph framework reported the worst results, because its architecture is not adapted for this type of data, or because it needs a large number of samples for the training of the ANN. The Neuroph framework reported better results with a large number of inputs for the ANN.

The use of the Encog framework slightly improved the results obtained with normalized data, reporting an accuracy of 37.07% using the dataset 2. However, Encog framework reported a high accuracy with the use of non-normalized data (74.45%). In contrast with the Neuroph framework, it was verified that the best accuracies were attained by the implementations with a smaller number of inputs.

The major problem of the implementation of DeepLearning4j framework is the resource consumption, where the performance is affected. However, the performance is only bad in the training phase. The final implementation the ANN provides reliable results after being trained. DeepLearning4j always reported high accuracy in the results with a large number of inputs—the results obtained were 80.35% accurate with non-normalized data, and 85.89% with normalized data.

The results recommend the DNN method with all features extracted from the acquired data as the most reliable method for the identification of ADLs. However, before its implementation, the data should be normalized with the mean and standard deviation method, and the L_2 regularization method should be applied. Based on the tests performed with the acquired data, the results obtained are always higher than those reported other ways. The results obtained have a *precision* value of 86.21%, a *recall* value of 85.89%, and an *F1 score* value of 86.05%.

In addition to the analysis, the confusion matrixes for the different frameworks were made, and are presented in Tables 3–8. By analyzing Table 3, it is possible to verify that the number of true positive values in recognition of walking upstairs, walking downstairs, and standing, is meager, proving a high number of false negatives and true negatives using the MLP method with the Neuroph framework based on non-normalized data. Next, Table 4 shows that the number of true positive values in recognition of all ADLs is meager, verifying a high number of false negatives using the MLP method with the Neuroph framework based on normalized data.

Following the analysis of Table 5, it was verified that only running is recognized by the MLP method with the Encog framework based on non-normalized data, presenting a high number of false negative values. In contrast, based on the implementation of the MLP method with the Encog framework based on normalized data, walking is always correctly recognized with 2000 true positive values, but it has 7999 false negative values. The high number of false negative values is also verified in the other ADLs, and the true negative and false positive values are too high.

Based on the use of the DNN method with the DeepLearning4j framework based on non-normalized data, the number of true negatives is only low in recognition of standing activity, reporting a high number of false positive values. However, the standing activity also reported a high number of true positive values, while the other ADLs reported high false negative values. Finally, with the use of the DNN method with the DeepLearning4j framework based on normalized data, the true positive and true negative values are high in all ADLs recognized.

Table 3. Confusion matrix of the results obtained with non-normalized data by the implementation of the MLP method with the Neuroph framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	2	3	1471	0	2000
True Negative	3474	3473	2005	3476	1476
False Positive	1998	1997	529	2000	0
False Negative	4526	4527	5995	4524	6524

Table 4. Confusion matrix of the results obtained with normalized data by the implementation of the MLP method with the Neuroph framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	0	0	162	0	200
True Negative	2162	2162	2000	2162	162
False Positive	2000	2000	1838	2000	0
False Negative	5838	5838	6000	5838	7838

Table 5. Confusion matrix of the results obtained with non-normalized data by the implementation of MLP method with Encog framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	0	0	1001	0	0
True Negative	1001	1001	0	1001	1001
False Positive	2000	2000	999	2000	2000
False Negative	6999	6999	8000	6999	6999

Table 6. Confusion matrix of the results obtained with normalized data by the implementation of the MLP method with the Encog framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	1	0	0	0	2000
True Negative	2000	2001	2001	2001	1
False Positive	1999	2000	2000	2000	0
False Negative	6000	5999	5999	5999	7999

Table 7. Confusion matrix of the results obtained with non-normalized data by the implementation of the DNN method with the DeepLearning4j framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	290	0	0	2000	0
True Negative	7786	7999	8000	506	7999
False Positive	214	1	0	7494	1
False Negative	1710	2000	2000	0	2000

Table 8. Confusion matrix of the results obtained with normalized data by the implementation of the DNN method with the DeepLearning4j framework.

	Walking Downstairs	Walking Upstairs	Running	Standing	Walking
True Positive	1334	1639	1909	1985	1722
True Negative	7641	7317	7978	7941	7712
False Positive	359	683	22	59	288
False Negative	666	361	91	15	278

This paper highlights the results obtained with different datasets using only the accelerometer data for the creation of a part of the method for the automatic recognition of several ADLs, including running, walking, walking upstairs and downstairs, and standing. The study also compares the results obtained with different types of ANNs, requiring low processing for the correct implementation in mobile devices.

The low accuracies verified with Neuroph and Encog frameworks are related to the fact that the ANNs created are probably overfitted. The possible solutions may be the acquisition of more data, the application of L_2 regularization, the implementation of dropout regularization, the early stopping of the training, the use of the batch normalization, or the use of a minor number of features in the ANN. The DNN method with L_2 regularization and normalized data reported the best results. The influence of the amount of the maximum iterations is not substantial, but, in some cases, it increases the accuracy of the ANN.

During the data acquisition, several constraints may exist, collecting noised values of sensors' data. Commonly, the accelerometer is available in all mobile devices, and the implementation of the system architecture for the recognition of ADLs and its environments can be possible with all devices in the market. However, these are multitasking devices, and sometimes the data cannot be collected or is incorrectly collected, providing low accuracy on the recognition of the ADL. Another example consists of the positioning of the mobile device because the data is not correctly acquired during a call. Memory and power processing are profoundly affected by the performance of different tasks at the same time.

The main focus of this research was to explore the use of the accelerometer sensor for ADLs recognition. We found that the accuracy obtained is in line with the previous results in the literature [20]. This study reports an accuracy of 85.89% in the recognition of five ADLs. Furthermore, using the

DNN method, according to Table 2, the results obtained with the implementation of our methods are not directly comparable, because the datasets and source code of the implementation used by other authors are not publicly available. A comparison would be essential to proving the reliability of our method. Thus, considering the average of the accuracies reported by ANNs and their variants shared in the literature, the results (92% ± 6.55%) present better accuracies than those obtained in this study. However, taking into account only the average of the accuracies reported by the projects that identified more than one ADL, the results reported by other studies (90% ± 6.60%) are slightly equivalent to those published by our research. Finally, considering only the studies that recognized five or more ADLs, the results reported by these studies (90% ± 6.63%) are equivalent to the results obtained with this work.

In conclusion, the accuracy of the ADLs recognition depends on several variables, including the conditions for data acquisition, conditions for data processing, and the use of lightweight methods (local processing) or server-side processing [72]. As presented in [72], it may cause failures on the data acquisition, collect incorrect data, or claim the nonexistence of data in some instances, causing improper recognition of ADL. To avoid some effects of inaccurate data, we implemented data cleaning methods, and data imputation methods may be useful for reducing the impacts of unavailable data. The main possible problems are related to the incorrect or nonexistent recognition of ADLs performed.

The main limitations of this study are related to the use of mobile devices for data acquisition. On the one hand, there is a lack of scientific evidence and research on the definition of the best position at which the mobile device must be located. On the other hand, other constraints during the data acquisition are related to the frequency of the data acquisition because it depends on the different processes running in the mobile device. During the experimental phase, the mobile application developed for the data acquisition writes the data in text files; the latency to write in the text files also influences the data acquisition and processing. However, the use of local processing and lightweight methods reduces the lag of the connection with the network, but the different methods must always be optimized.

Taking into account the results obtained in [43], the number of ADLs recognized, the number of records for each ADL, and the features extracted are different in our study. Consequently, the accuracy obtained in our research with the DNN method is higher than the results reported by the authors of [43]. We expect that in similar conditions of study [43], we obtain the same or better results. Nevertheless, it will be impossible to test, as the authors [43] did not make their data publicly available.

5. Conclusions

This paper presents several approaches that use the accelerometer sensor commonly available in mobile devices for ADLs recognition. Furthermore, the main contribution of this document is to offer a comparative study of different ANN implementations to find the most appropriate method for ADLs identification using only accelerometer data. The comparative study performed in this research recommends the use of DNN for the recognition of ADLs. We proposed the implementation of the trained DNN method in the system for the identification of the ADLs using only the accelerometer sensor available in off-the-shelf mobile devices, applied with the DeepLearning4j framework. The results show the accuracy of 85.89%, a *precision* value of 86.21%, a *recall* value of 85.89%, and an *F1 score* value of 86.05% using the five largest distances between the maximum peaks; the mean, standard deviation, variance, and median of the maximum peaks; and the standard deviation, mean, maximum and minimum values, variance, and median of the raw signal as features.

Nevertheless, this study has some limitations concerning the use of mobile devices. The lack of research on the best position of the mobile device for data collection is a relevant question. Moreover, the energy expenditure concerning the processing power related to the frequency of data acquisition is also a significant challenge that the authors have addressed by using only accelerometer data. The authors verified that the overfitting problem is not avoided, but the results obtained using only accelerometer data are similar to those obtained with the use of multiple sensors. Additionally, the authors found that using only one sensor and a smaller number of features for the train of the ANN does not significantly decrease the accuracy of the results obtained. Still, it uses less computational resources and promotes the energy consumption of the mobile device when compared with the use of multiple sensors.

As future work, other implementation settings regarding different machine learning methods will be studied. These implementations will include the design of other types of data classification methods, *e.g.*, ensemble learning methods and decision trees, to verify the existence of different approaches with better results using our dataset. The dataset is publicly available, and other authors can use and compare it with their methods.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, writing—original draft preparation, and writing—review and editing: I.M.P., G.M., N.M.G., F.F.-R., M.C.T., E.Z., S.S., and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by FCT/MCTES through national funds, and when applicable, co-funded EU funds under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável cofinanciado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020*).

Acknowledgments: This work is funded by FCT/MCTES through national funds, and when applicable, co-funded EU funds under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável cofinanciado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020*). This article/publication is based on work from COST Action IC1303: AAPELE—Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA16226; SHELD-ON—Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information at www.cost.eu. COST (European Cooperation in Science and Technology) is a funding agency for research and innovation networks. Our Actions help connect research initiatives across Europe and enable scientists to grow their ideas by sharing them with their peers. This boosts their research, career and innovation.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Salazar, L.H.A.; Lacerda, T.; Nunes, J.V.; von Wangenheim, C.G. A systematic literature review on usability heuristics for mobile phones. *Int. J. Mob. Hum. Comput. Interact. (IJMHCI)* **2013**, *5*, 50–61. [CrossRef]
2. Marques, G. Ambient Assisted Living and Internet of Things. In *Harnessing the Internet of Everything (IoE) for Accelerated Innovation Opportunities*; IGI Global: Hershey, PA, USA, 2019; p. 100. [CrossRef]
3. Pedretti, L.W.; Early, M.B. *Occupational Therapy: Practice Skills for Physical Dysfunction*; Mosby: St. Louis, MO, USA, 2001.
4. Garcia, N.M. A roadmap to the design of a personal digital life coach. In *International Conference on ICT Innovations*; Springer: Berlin, Germany, 2015; pp. 21–27.
5. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. From Data Acquisition to Data Fusion: A Comprehensive Review and a Roadmap for the Identification of Activities of Daily Living Using Mobile Devices. *Sensors* **2016**, *16*, 184. [CrossRef] [PubMed]
6. Pires, I.M.; Garcia, N.M.; Flórez-Revuelta, F. Multi-sensor data fusion techniques for the identification of activities of daily living using mobile devices. In *Proc European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - ECML/PKDD*; CEUR: Porto, Portugal, 2015.
7. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Identification of Activities of Daily Living Using Sensors Available in off-the-shelf Mobile Devices: Research and Hypothesis. In *Ambient Intelligence—Software and Applications—7th International Symposium on Ambient Intelligence (ISAmI 2016)*; Lindgren, H., De Paz, J.F., Novais, P., Fernández-Caballero, A., Yoe, H., Jiménez Ramírez, A., Villarrubia, G., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 121–130.
8. Marques, G.; Pitarma, R.; Garcia, N.M.; Pombo, N. Internet of Things Architectures, Technologies, Applications, Challenges, and Future Directions for Enhanced Living Environments and Healthcare Systems: A Review. *Electronics* **2019**, *8*, 81. [CrossRef]
9. Akhoundi, M.A.A.; Valavi, E. Multi-sensor fuzzy data fusion using sensors with different characteristics. *arXiv* **2010**, arXiv:1010.6096.

10. Banos, O.; Damas, M.; Pomares, H.; Rojas, I. On the Use of Sensor Fusion to Reduce the Impact of Rotational and Additive Noise in Human Activity Recognition. *Sensors* **2012**, *12*, 8039–8054. [CrossRef] [PubMed]
11. Dernbach, S.; Das, B.; Krishnan, N.C.; Thomas, B.L.; Cook, D.J. Simple and Complex Activity Recognition through Smart Phones. In Proceedings of the 2012 Eighth International Conference on Intelligent Environments, Guanajuato, Mexico, 26–29 June 2012; pp. 214–221. [CrossRef]
12. Hsu, Y.; Chen, K.; Yang, J.; Jaw, F. Smartphone-based fall detection algorithm using feature extraction. In Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15–17 October 2016; pp. 1535–1540. [CrossRef]
13. Paul, P.; George, T. An effective approach for human activity recognition on smartphone. In Proceedings of the 2015 IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, India, 20 March 2015; pp. 1–3. [CrossRef]
14. Shen, C.; Chen, Y.; Yang, G. On motion-sensor behavior analysis for human-activity recognition via smartphones. In Proceedings of the 2016 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), Sendai, Japan, 29 February 29–2 March 2016; pp. 1–6. [CrossRef]
15. Doya, K.; Wang, D. Exciting Time for Neural Networks. *Neural Netw.* **2015**, *61*, xv–xvi. [CrossRef]
16. Wang, D. Pattern recognition: neural networks in perspective. *IEEE Expert* **1993**, *8*, 52–60. [CrossRef]
17. Neuroph. 2019. Available online: http://neuroph.sourceforge.net/ (accessed 20 March 2019).
18. Encog. 2017. Available online: http://www.heatonresearch.com/encog/ (accessed 20 March 2019).
19. Deeplearning4j. 2019. Available online: https://deeplearning4j.org/ (accessed 20 March 2019).
20. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C. Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices. *Pervasive Mob. Comput.* **2018**, *47*, 78–93. [CrossRef]
21. Zdravevski, E.; Lameski, P.; Trajkovik, V.; Kulakov, A.; Chorbev, I.; Goleva, R.; Pombo, N.; Garcia, N. Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering. *IEEE Access* **2017**, *5*, 5262–5280. [CrossRef]
22. Gadebe, M.L.; Kogeda, O.P.; Ojo, S.O. Personalized Real Time Human Activity Recognition. In Proceedings of the 2018 5th International Conference on Soft Computing Machine Intelligence (ISCMI), Nairobi, Kenya, 21–22 November 2018; pp. 147–154. [CrossRef]
23. Naved, M.M.A.; Uddin, M.Y.S. Adaptive Notifications Generation for Smartphone Users Based on their Physical Activities. In Proceedings of the 2018 5th International Conference on Networking, Systems and Security (NSysS), Dhaka, Bangladesh, 18–20 December 2018; pp. 1–9. [CrossRef]
24. RoyChowdhury, I.; Saha, J.; Chowdhury, C. Detailed Activity Recognition with Smartphones. In Proceedings of the 2018 Fifth International Conference on Emerging Applications of Information Technology (EAIT), Kolkata, India, 12–13 January 2018, pp. 1–4. [CrossRef]
25. Sukor, A.S.A.; Zakaria, A.; Rahim, N.A. Activity recognition using accelerometer sensor and machine learning classifiers. In Proceedings of the 2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA), Batu Feringghi, Malaysia, 9–10 March 2018; pp. 233–238. [CrossRef]
26. Yan, N.; Chen, J.; Yu, T. A Feature Set for the Similar Activity Recognition Using Smartphone. In Proceedings of the 2018 10th International Conference on Wireless Communications and Signal Processing (WCSP), Hangzhou, China, 18–20 October 2018; pp. 1–6. [CrossRef]
27. Lavanya, B.; Gayathri, G.S. Exploration and Deduction of Sensor-Based Human Activity Recognition System of Smart-Phone Data. In Proceedings of the 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Coimbatore, India, 14–16 December 2017; pp. 1–5. [CrossRef]
28. Li, G.; Huang, L.; Xu, H. iWalk: Let Your Smartphone Remember You. In Proceedings of the 2017 4th International Conference on Information Science and Control Engineering (ICISCE), Changsha, China, 21–23 July 2017; pp. 414–418. [CrossRef]
29. Tsinganos, P.; Skodras, A. A smartphone-based fall detection system for the elderly. In Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis, Ljubljana, Slovenia, 18–20 September 2017; pp. 53–58. [CrossRef]
30. Wannenburg, J.; Malekian, R. Physical Activity Recognition From Smartphone Accelerometer Data for User Context Awareness Sensing. *IEEE Trans. Syst. Man, Cybern. Syst.* **2017**, *47*, 3142–3149. [CrossRef]

31. Cardoso, N.; Madureira, J.; Pereira, N. Smartphone-based transport mode detection for elderly care. In Proceedings of the 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), Munich, Germany, 14–17 September 2016; pp. 1–6. [CrossRef]
32. Dangu Elu Beily, M.; Badjowawo, M.D.; Bekak, D.O.; Dana, S. A sensor based on recognition activities using smartphone. In Proceedings of the 2016 International Seminar on Intelligent Technology and Its Applications (ISITIA), Lombok, Indonesia, 28–30 June 2016; pp. 393–398. [CrossRef]
33. Sen, S.; Rachuri, K.K.; Mukherji, A.; Misra, A. Did you take a break today? Detecting playing foosball using your smartwatch. In Proceedings of the 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), Sydney, Australia, 14–18 March 2016; pp. 1–6. [CrossRef]
34. Weiss, G.M.; Lockhart, J.W.; Pulickal, T.T.; McHugh, P.T.; Ronan, I.H.; Timko, J.L. Actitracker: A Smartphone-Based Activity Recognition System for Improving Health and Well-Being. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 17–19 October 2016; pp. 682–688. [CrossRef]
35. Wang, C.; Xu, Y.; Zhang, J.; Yu, W. SW-HMM: A Method for Evaluating Confidence of Smartphone-Based Activity Recognition. In Proceedings of the 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, China, 23–26 August 2016; pp. 2086–2091. [CrossRef]
36. Guo, H.; Chen, L.; Chen, G.; Lv, M. An Interpretable Orientation and Placement Invariant Approach for Smartphone Based Activity Recognition. In Proceedings of the 2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, 10–14 August 2015; pp. 143–150. [CrossRef]
37. Kim, Y.; Kang, B.; Kim, D. Hidden Markov Model Ensemble for Activity Recognition Using Tri-Axis Accelerometer. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 9–12 October 2015; pp. 3036–3041. [CrossRef]
38. Kwon, Y.; Kang, K.; Bae, C. Analysis and evaluation of smartphone-based human activity recognition using a neural network approach. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–16 July 2015; pp. 1–5. [CrossRef]
39. Ling, Y.; Wang, H. Unsupervised Human Activity Segmentation Applying Smartphone Sensor for Healthcare. In Proceedings of the 2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, 10–14 August 2015; pp. 1730–1734. [CrossRef]
40. Torres-Huitzil, C.; Nuno-Maganda, M. Robust smartphone-based human activity recognition using a tri-axial accelerometer. In Proceedings of the 2015 IEEE 6th Latin American Symposium on Circuits Systems (LASCAS), Montevideo, Uruguay, 24–27 February 2015; pp. 1–4. [CrossRef]
41. Wang, C.; Zhang, W. Activity Recognition Based on Smartphone and Dual-Tree Complex Wavelet Transform. In Proceedings of the 2015 8th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 12–13 December 2015; Volume 2, pp. 267–270. [CrossRef]
42. Zainudin, M.N.S.; Sulaiman, M.N.; Mustapha, N.; Perumal, T. Activity recognition based on accelerometer sensor using combinational classifiers. In Proceedings of the 2015 IEEE Conference on Open Systems (ICOS), Bandar Melaka, Malaysia, 24–26 August 2015; pp. 68–73. [CrossRef]
43. Zhang, L.; Wu, X.; Luo, D. Real-Time Activity Recognition on Smartphones Using Deep Neural Networks. In Proceedings of the 2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, 10–14 August 2015; pp. 1236–1242. [CrossRef]
44. Aguiar, B.; Silva, J.; Rocha, T.; Carneiro, S.; Sousa, I. Monitoring physical activity and energy expenditure with smartphones. In Proceedings of the IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), Hong Kong, China, 5–7 January 2014; pp. 664–667. [CrossRef]
45. Fahim, M.; Lee, S.; Yoon, Y. SUPAR: Smartphone as a ubiquitous physical activity recognizer for u-healthcare services. In Proceedings of the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, 26–30 August 2014; pp. 3666–3669. [CrossRef]

46. Khalifa, S.; Hassan, M.; Seneviratne, A. Feature selection for floor-changing activity recognition in multi-floor pedestrian navigation. In Proceedings of the 2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU), Singapore, 6–8 January 2014; pp. 1–6. [CrossRef]
47. Duarte, F.; Lourenço, A.; Abrantes, A. Activity classification using a smartphone. In Proceedings of the 2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013), Lisbon, Portugal, 9–12 October 2013; pp. 549–553. [CrossRef]
48. Fan, L.; Wang, Z.; Wang, H. Human Activity Recognition Model Based on Decision Tree. In Proceedings of the 2013 International Conference on Advanced Cloud and Big Data, Nanjing, China, 13–15 December 2013; pp. 64–68. [CrossRef]
49. Lau, S.L. Comparison of orientation-independent-based-independent-based movement recognition system using classification algorithms. In Proceedings of the 2013 IEEE Symposium on Wireless Technology Applications (ISWTA), Kuching, Malaysia, 22–25 September 2013; pp. 322–326. [CrossRef]
50. Mitchell, E.; Monaghan, D.; O'Connor, N.E. Classification of Sporting Activities Using Smartphone Accelerometers. *Sensors* **2013**, *13*, 5317–5337. [CrossRef] [PubMed]
51. Oshin, T.O.; Poslad, S. ERSP: An Energy-Efficient Real-Time Smartphone Pedometer. In Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; pp. 2067–2072. [CrossRef]
52. Bujari, A.; Licar, B.; Palazzi, C.E. Movement pattern recognition through smartphone's accelerometer. In Proceedings of the 2012 IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, 14–17 January 2012; pp. 502–506. [CrossRef]
53. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity Recognition Using Cell Phone Accelerometers. *SIGKDD Explor. Newsl.* **2011**, *12*, 74–82. [CrossRef]
54. Lau, S.L.; David, K. Movement recognition using the accelerometer in smartphones. In Proceedings of the 2010 Future Network Mobile Summit, Florence, Italy, 16–18 June 2010; pp. 1–9.
55. Lau, S.L.; König, I.; David, K.; Parandian, B.; Carius-Düssel, C.; Schultz, M. Supporting patient monitoring using activity recognition with a smartphone. In Proceedings of the 2010 7th International Symposium on Wireless Communication Systems, York, UK, 19–22 September 2010; pp. 810–814. [CrossRef]
56. Stenneth, L.; Wolfson, O.; Yu, P.S.; Xu, B. Transportation mode detection using mobile phones and GIS information. In Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems, Chicago, IL, USA, 1–4 November 2011; pp. 54–63.
57. Borazio, M.; Van Laerhoven, K. Using time use with mobile sensor data: a road to practical mobile activity recognition? In Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia, Lulea, Sweden, 2–5 December 2013; pp. 1–10.
58. Hong, J.H.; Ramos, J.; Shin, C.; Dey, A.K. An activity recognition system for ambient assisted living environments. In *International Competition on Evaluating AAL Systems Through Competitive Benchmarking*; Springer: Berlin, Germany, 2012; pp. 148–158.
59. Ignatov, A.D.; Strijov, V.V. Human activity recognition using quasiperiodic time series collected from a single tri-axial accelerometer. *Multimed. Tools Appl.* **2016**, *75*, 7257–7270. [CrossRef]
60. Khan, A.M.; Siddiqi, M.H.; Lee, S.W. Exploratory data analysis of acceleration signals to select light-weight and accurate features for real-time activity recognition on smartphones. *Sensors* **2013**, *13*, 13099–13122. [CrossRef] [PubMed]
61. Pereira, J.D.; da Silva e Silva, F.J.; Coutinho, L.R.; de Tácio Pereira Gomes, B.; Endler, M. A movement activity recognition pervasive system for patient monitoring in ambient assisted living. In Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, 4–8 April 2016; pp. 155–161.
62. Torres-Huitzil, C.; Alvarez-Landero, A. Accelerometer-based human activity recognition in smartphones for healthcare services. In *Mobile Health*; Springer: Berlin, Germany, 2015; pp. 147–169.
63. Tao, D.; Wen, Y.; Hong, R. Multicolumn bidirectional long short-term memory for mobile devices-based human activity recognition. *IEEE Internet Things J.* **2016**, *3*, 1124–1134. [CrossRef]
64. Zdravevski, E.; Risteska Stojkoska, B.; Standl, M.; Schulz, H. Automatic machine-learning based identification of jogging periods from accelerometer measurements of adolescents under field conditions. *PLoS ONE* **2017**, *12*, e0184216. [CrossRef] [PubMed]

65. Github. Impires/August_2017-_Multi-Sensor_Data_Fusion_in_Mobile_Devices_for_the_Identification_of_Activities_of_Dail. 2018. Available online: https://github.com/impires/August_2017-_Multi-sensor_data_fusion_in_mobile_devices_for_the_identification_of_activities_of_dail (accessed 20 March 2019).
66. BQ. Smartphones BQ Aquaris | BQ Portugal. 2019. Available online: https://www.bq.com/pt/smartphones (accessed 20 March 2019).
67. Graizer, V. Effect of low-pass filtering and re-sampling on spectral and peak ground acceleration in strong-motion records. In Proceedings of the 15th World Conference of Earthquake Engineering, Lisbon, Portugal, 24–28 September 2012; pp. 24–28.
68. Lameski, P.; Zdravevski, E.; Koceski, S.; Kulakov, A.; Trajkovik, V. Suppression of Intensive Care Unit False Alarms Based on the Arterial Blood Pressure Signal. *IEEE Access* **2017**, *5*, 5829–5836. [CrossRef]
69. Hajela, P.; Berke, L. Neural networks in structural analysis and design: an overview. *Comput. Syst. Eng.* **1992**, *3*, 525–538. [CrossRef]
70. prateekvjoshi. Understanding Xavier Initialization in Deep Neural Networks. 2016. Available online: https://prateekvjoshi.com/2016/03/29/understanding-xavier-initialization-in-deep-neural-networks/ (accessed 20 March 2019).
71. Ng, A.Y. Feature Selection, L1 vs. L2 Regularization, and Rotational Invariance. In *ICML '04, Proceedings of the Twenty-first International Conference on Machine Learning, Banff, AB, Canada, 4–8 July 2004*; ACM: New York, NY, USA, 2004; p. 78. [CrossRef]
72. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Limitations of the Use of Mobile Devices and Smart Environments for the Monitoring of Ageing People. In *Proceedings of the 4th International Conference on Information and Communication Technologies for Ageing Well and e-Health - Volume 1: HSP*; Science and Technology Publications: Setúbal, Portugal, 2018; pp. 269–275. ISBN 978-989-758-299-8. [CrossRef]

electronics

MDPI

Article

A Low-Cost Cognitive Assistant

Angelo Costa [1,*], Jaime A. Rincon [2], Vicente Julian [2] and Paulo Novais [1] and Carlos Carrascosa [2]

[1] ALGORITMI Centre, University of Minho, 4710-057 Braga, Portugal; pjon@di.uminho.pt
[2] Institut Valencià d'Investigació en Intel·ligència Artificial (VRAIN), Universitat Politècnica de València, 46022 València, Spain; jrincon@dsic.upv.es (J.A.R.); vinglada@dsic.upv.es (V.J.); carrasco@dsic.upv.es (C.C.)
* Correspondence: acosta@di.uminho.pt

Received: 29 December 2019; Accepted: 5 February 2020; Published: 11 February 2020

Abstract: In this paper, we present in depth the hardware components of a low-cost cognitive assistant. The aim is to detect the performance and the emotional state that elderly people present when performing exercises. Physical and cognitive exercises are a proven way of keeping elderly people active, healthy, and happy. Our goal is to bring to people that are at their homes (or in unsupervised places) an assistant that motivates them to perform exercises and, concurrently, monitor them, observing their physical and emotional responses. We focus on the hardware parts and the deep learning models so that they can be reproduced by others. The platform is being tested at an elderly people care facility, and validation is in process.

Keywords: cognitive assistants; aging; emotion recognition

1. Introduction

We are currently facing a societal problem: the world's population is growing older fast [1]. While this could be great news (and it is), there are intrinsic problems that come with fast shifting demographic changes, and being unprepared for growing health requirements that elderly people have is one of them.

Persons over the age of 65 are the fastest-growing age group, and it is expected that by 2050, 16% of the world population will be over 65 years, while in 2019, this value was already 9% [1]. This projection is global, meaning that regions such as Northern Africa and Western Asia, Central and Southern Asia, Eastern and South-Eastern Asia, and Latin America and the Caribbean are expected to double their elderly population [1]. Furthermore, by 2050, 25% of the population in Europe will be 65 years or over, being accompanied by an interesting fact: in 2018, children under five years of age were outnumbered by persons aged over 65 [1]. This rapid increase is mainly due to improved medical care, which diminishes the mortality rate. Although people live longer, this is not without its problems. In the Euro28 area, it is expected that people over 65 years only have 10 more years (on average) until serious health problems start to appear, as reported by the United Nations (UN) [1].

In their latest census, the UN has identified that there is an increasing shortage of employed people, thus causing high stress to social protection systems [1]. This is due to two factors: the decrease of working-age people and the social-economic problems of countries. For instance, in Japan, the ratio between people aged 25–64 to those over age 65 is 1.8, while in most of Europe, the value is starting to fall below of three. This means that there will be a high impact to countries' economies as the GDP will be affected by the decrease of the labor market, being overburdened by the increasing costs of healthcare systems, pensions, and social protection.

Apart from the economic distress, there is the healthcare distress. Studies, like the ones that were presented by Licher [2] and Jaul [3], show that maintaining high levels of quality of life while aging is complicated. There is no clear path towards a definitive medical solution, as most of the illnesses are non-curable and have very complex pathologies. A possible stand-in replacement to a medical treatment is backed by several research studies [4–6] that show that it is possible to refrain from the advances of these illnesses by keeping the elderly active both physically and cognitively through exercises. These exercises are often complex and require attention from a caregiver, either to help perform those exercises or to correct the posture/strategy. This requires a large amount of monitoring time by another person.

The caregivers are often overburdened by the care and assistance work, as evidenced by these works [7–9] (with the most severe cases being non-specialized trained caregivers combined with high-dependency elders). This means that often, the caregiver puts his/her health at risk, and the elder does not receive adequate care. In specialized facilities, like nursing homes, this is less accentuated; nonetheless, the lack of caregivers and the high number of residents may lead to a poorer experience in these facilities [10]. Furthermore, as previously stated, the elderly are not economically prepared for the high cost that these facilities charge [11].

In short, in the near future, a large number of older people will be left alone in their homes while suffering from limiting and life-threatening diseases because they cannot afford nursing homes or home care services or because they are not able to have assistance from an informal caregiver.

A possible solution to these issues may be the usage of technology to help elderly people perform Activities of Daily Living (ADL) or attenuate their loneliness, while actively monitoring their health status. There are already some projects in this domain, explained in depth in Section 2. These projects shed some light on what the current approaches are, and more importantly, what the needs of elderly people are and how technology can improve their quality of life.

Our project goal is to create a way so that elderly people can in stay their homes safely and under active supervision, while at the same time engaging them in personalized active exercises and exergames. The way that this goal is achieved by our platform is by using a low-cost, easy-to-deploy sensor system that is able to monitor said exercises and interact with the elders, sending reports to the informal/formal caregivers. The platform is constituted by two components: the sensors and the software (exercise evaluator, scheduler, information portal, and interactor). Our proposal covers a less traveled path, which is the usage of low-cost sensors (using in expensive commercial components and 3D printing) together with health-related software that gives personalized advice.

This paper is a continuation of the work presented in [12]. The main improvements in relation to it are the improved sensor systems and the learning models for information extraction. Additionally, the objective is that others are able to reproduce our platform with ease from the information presented in this paper.

The paper is structured as follows. Section 2 analyses the related work. Section 3 presents the proposed system, which describes the hardware (wristbands) and the emotion and activity detection. Finally, the conclusions are presented in Section 4.

2. Related Work

This project touches on prolific domains: emotion detection, human activity recognition, and cognitive assistants. Therefore, in this section, we present related work that belongs to those domains or, like this project, touches on all or part of them.

2.1. Cognitive Assistants

The cognitive assistants consist typically of a combination of software and hardware systems that help people (mostly cognitively impaired) in their ADL. The aim is to provide memory assistance (through reminders), visual/auditory cues, and physical assistance (through robots or smart home actuators) [13,14].

One example of this is the PHAROS [15] project, whose goal is to use a friendly-looking robot to engage elderly people in playful activities, such as physical exercises or cognitive games. The aim is to maintain a conversation that subliminally engages the users to perform the system’s suggestions. Furthermore, using the robot sensors, it is able to detect and gauge the exercise performance and give this information to the caregivers so they are able to access if the users are performing the exercise well and to measure if the users are losing abilities.

Using friendly robots to interact with the user was the work of Castillo et al. [16]. The objective was to use a robot to guide the users in therapy sessions for apraxia of speech. The robot captured the mouth movements and evaluated if they were correct, giving the users advice on how to perform the exercises and tips about mouth positions.

The CoMEproject [17] is an example of a cognitive assistant that does have a robot counterpart. This project uses wearable sensors and smartphones to monitor the users, giving way to interaction while concurrently collecting reports from the usage. The users receive information and have tutorials available on how to perform the planned activities. The caregivers are able to access the user performance reports. This project is designed to be implemented in elderly care facilities, maximizing the number of care receivers a caregiver is able to monitor.

The iGenda project [18–20] aims to provide assistance through ambient assisted living devices/environments like a smart home. The objective is to use IoT or Internet connected devices to convey information and actuators to change the environments for the users. The social objective is to be a cognitive aid to people who are suffering from light to mild cognitive disabilities. iGenda's core is an event management system that monitors the users' tasks and shared activities and provides cues through screens and speakers to remind the users of the upcoming activities. Furthermore, the users are able to interact with iGenda, using logical arguments and persuading them to perform certain activities. Apart from this, iGenda is able to monitor users outside their home, resorting to information of their smartphone; thus, it is able to verify if they are leaving safe/common areas.

2.2. Human Activity Recognition

The domain of human activity recognition is experiencing a boom in terms of development due to the usage of novel deep learning techniques that were not available previously. Several studies [21,22] showed that the majority of current projects and technologies used in human activity recognition display a clear pattern: deep learning and datasets. This pattern allows the advancement of the developments to the stage of micro-optimization due most models having over 85% accuracy.

One example is the work of Martinez-Martin et al. [23–25], which proposed a rehabilitation system to provide rehabilitation monitoring at home using a humanoid robot. The goal was to use the robot's cameras to access the user's physical movements visually, using deep learning methods, and correct them using the robot screen and body to convey this information. The robot was also able to navigate around the house and locate the user. The captured information (body movement measure) was made available to healthcare professionals for them to correct the user if needed, providing specialized attention.

The work of Vepakomma et al. [26] presented a framework that detected common home activities from wrist bracelets. They resorted to deep learning methods to classify the raw input and produce a result from even light gestures. Their framework was able to detect 22 distinct activities with an accuracy of 90%. The issue with this project was that it was too personalized, meaning that these results were achieved with only two persons, whereas the results were significantly lower with others users.

The work of Cao et al. [27] presented a novel classification method that achieved over 94% accuracy in detecting ADL. The method worked by creating associations between activities determining how usual a sequence of events was, like rinsing the mouth with water performed after brushing teeth. Using these pre-established associations was faster than calculating real-time data. The downside of this approach was its rigidity to changes and that singular activities were harder to detect, apart from being required to input these associations by a technician, as the system was unable to learn on its own.

2.3. Emotion Detection

A novel domain is emotion detection, where, using a combination of hardware and software, computer systems are able to identify human emotions. Several studies reported that there were various methods to human emotion recognition [28,29]. There was a division between using non-invasive sensors (like vital signs sensors) and using cameras. We focused on the advancements of detection using body sensors, as used in this project. This decision was based on the privacy issues arising from using cameras.

Brás et al. [30] presented 90% accuracy in detecting emotions using Electrocardiogram (ECG) sensors, in a controlled environment. To achieve this high result, a novel approach was developed, using a quantization method that compared the incoming signal to a dataset doing a meta-classification; then compressing ECG meta-data resorting to an ECG dataset as a reference; finally, using the probability that the ECG was classified correctly. This unorthodox process was limited to a tight coupling of the models to the individuals that were used to train the system. The tests may have introduced a bias in the results; for instance, it was reasonable to assume that people became scared and anxious when they were exposed to fearful situations. Fear is an intense emotion that regularly leads to an accelerated heartbeat, which is simple to identify in an ECG. The studies performed were designed to cause a strong emotional response, the minimum threshold values being unknown and whether muted emotions could be detected.

Using the matching pursuit algorithm and a probabilistic neural network method, Goshvarpour et al. [31] detected emotional features using ECG and Galvanic Skin Response (GSR). Nonetheless, in this work, only four emotions were detected: scary, happy, sad, and peaceful (from the pleasure arousal dominance model). As a trigger, music was used on eleven students. Over 90% accuracy was reported. From the study, it was determined that GSR had little impact on emotion detection. Furthermore, the emotions were not linearly detected. Strong emotions, like arousal (happy), were far simpler to detect than the others.

Naji et al. [32,33] used a combination of ECG with forehead biosignals to obtain a good accuracy in emotion identification. It was discovered that facial movements (like frowning) were very useful to identify emotions accurately. With the usage of the headband, a camera was not needed; thus, the privacy concerns were not significant.

Seoane et al. [34] used body sensors to detect stress levels of military personnel (ATRECproject). They established that placing the sensors (ECG and GSR) on the neck (throat area) provided a high level of accuracy in terms of valence markers and alert levels, which are directly related to stress levels. On the contrary, speech, GSR (on the hands/arms), or skin temperature provided little accuracy for emotion detection.

As can be seen, there are different (even contradictory) approaches to classifying emotions with minimal intrusion. ECG is crucial for the detection and classification of emotions, and the use of various sensors can improve the accuracy of the classification or help to detect triggering events.

With this project, we aim at the advancement of the state-of-the-art, by overcoming the issues that the projects presented in this section had. Nonetheless, it is of note that these projects were important hallmarks and should be regarded as so, as they established the pathway to newer advancements.

3. Low-Cost Cognitive Assistant

This section describes our proposal for a system that is a continuation of previous research presented in [12]. This new research incorporated a series of devices capable of detecting and classifying the movements carried out by elderly people and detecting their emotions when performing them.

With the emergence of wearable devices capable of counting daily steps and calculating the Heart Rate (HR), the use of these devices has many fields of application, the most common being in sport. Nevertheless, many healthcare related applications have emerged using these devices. Devices such as the Fitbit (https://www.fitbit.com/es/home) [35], which can be used to track physical activity, or the

Apple Watch [36], which can be used to monitor people with cardiovascular diseases (through heart rate measurements), are some of the examples in which these devices are used.

In recent years, new devices have appeared including communication protocols such as WiFi and Bluetooth. All these features are used to create applications that facilitate the monitoring of the elderly, allowing the acquisition of signals such as ECG, Photoplethysmography (PPG), respiratory rate, and GSR.

Our device was designed by integrating two elements, the emotion detection using bio-signals and the detection of movements in the lower and upper extremities through accelerometers.

To make this application possible, it was necessary to use different types of hardware that facilitated the acquisition of data and software tools that analyzed the information sent by the devices. This way, mixing these technologies, it was possible to recognize patterns, analyze images, analyze emotions, detect stress, etc.

3.1. Wristbands

We devised a set of two wristband prototypes as shown in Figure 1 to be worn by the people being monitored. Wristband B detected motion, whilst Wristband A had a more complex composition, as can be observed in Figure 2. The goal pursued by using both of these wristbands was to detect not only the emotion of the people being monitored, but also if they were properly doing the exercises being suggested. The decision to manufacture our own devices was due to the fact that the data from the commercial wristbands were filtered and preprocessed, so they did not have the precision required for our platform.

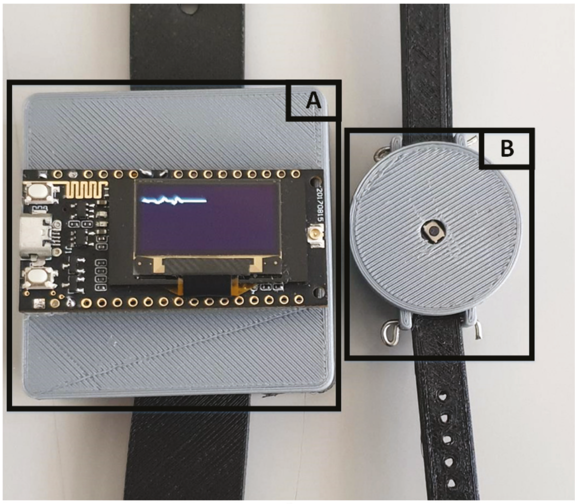


Figure 1. Wristband A and Wristband B (motion detector) prototypes.

To perform the emotion detection using bio-signals, it was necessary to have a specific hardware to acquire these signals. We designed a device capable of acquiring these signals so we could control the tuning and raw signal. There were two signals captured by our device, and the first was a PPG signal. This measurement was made by a sensor (Figure 3) that passed a light beam over the skin, to make the subcutaneous vessels illuminate. This made a part of this beam be reflected, falling on a photo sensor that converted it into an equivalent voltage. Because the skin absorbed more than 90% of the light, the diode pair was accompanied by amplifiers and filters that ensured an adequate voltage.

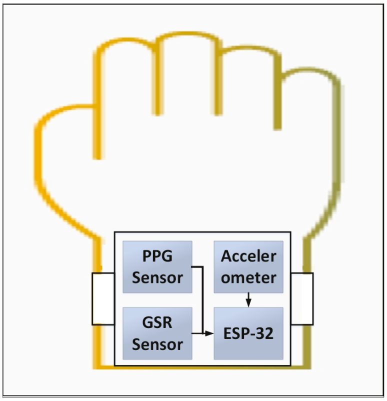
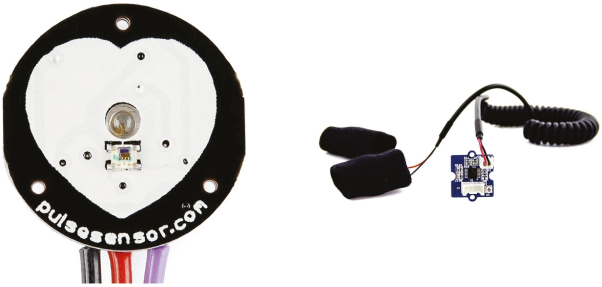


Figure 2. Wristband A prototype composition.



(**a**) PPG sensor (**b**) GSR sensor

Figure 3. Photoplethysmography (PPG) and Galvanic Skin Response (GSR) sensors.

The second signal captured by our system was skin resistance, which is the galvanic response of the skin. This resistance varies with the state of the skin's sweat glands, which are regulated by the Autonomous Nervous System (ANS). If the sympathetic branch of the ANS is excited, the sweat glands increase their activity by modifying the conductance of the skin. The ANS is directly related to the regulation of emotional behavior in human beings. To capture these variations, a series of electronic devices was used, equipped with sensors or electrodes that were in contact with the skin. When there was a variation in skin resistance, these devices registered this activity and returned an analog signal, which was proportional to the activity of the skin. Figure 3 shows the device used to make this capture.

The analog signals returned by the sensors were digitized using the ESP-32's analog to digital converter. Our system used an ESP-32 TTGO development system (Figure 4), which is being widely used in IoT applications. This was mainly due to its easy programming and to the fact that it had WiFi, LoRa, and Bluetooth communication protocols with low power consumption or BLE. These features make this device the ideal tool to be used in monitoring applications.

In this way, the ESP-32 transformed the analog signals returned by the sensors to digital. This was done using the analog-to-digital converter of the ESP-32. The digitized signals were transformed as voltage equivalent to the acquired signal. To carry out the transmission of the acquired data, one of the communication protocols incorporated in the development system was used. The ESP-32 TTGO incorporated three communication protocols, WiFi, LoRa, and low power Bluetooth. We used the HTTP protocol for data transfer via WiFi to the server.

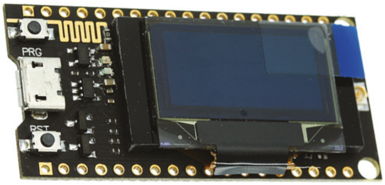


Figure 4. ESP-32 TTGO developer board.

3.2. Software

The sensor systems provided data about the human condition, and with this data we could form information about the exercises that the users performed and their emotional condition. This was performed by two modules that identified the exercises' performance and the emotions. Additionally, this information was made available to the users and caregivers so they were informed about their progression.

3.2.1. Emotion Classification

To perform the emotion detection using biosignals, it was necessary to calculate the biosignal values corresponding to each emotion for concrete individuals, as biosignals vary for each person. Therefore, a dataset was created to train an artificial neural network that gave us the emotion values of each individual using the biosignals as input.

The experiment to create the dataset acquired the signals of GSR and PPG while observing a series of images, which sought to modify our emotions [37]. The experiments were performed by 20 test subjects using a database with 1182 images. This database was divided into two sets: the training set of 900 images and the test set of 282 images.

Each experiment was composed by the following steps:

1. The set of training images (50) was observed by the test subjects for 10 s. During these 10 s, the signals of GSR, PPG, temperature, and heart rate were recorded and stored.
2. At the same time, the subject was recorded using a camera in the monitoring system. The images recorded were used to detect the emotion expressed by the subject. This detection was performed using the Microsoft Detect Emotions Service, which detects the following emotions: anger, contempt, disgust, fear, happiness, neutral, sadness, and surprise.
3. After 10 s of observation of the stimulus, the subject had 10 s more to respond to the SAM (Self-Assessment Manikin) test [38]. This test allowed us to know the emotional state of the individual in terms of PAD (Pleasure, Arousal, Dominance).

However, our dataset had two outputs: the first one corresponding to the emotion detected using the image processing and the second one the emotion obtained using the SAM (Self-Assessment Manikin) test [39]. The SAM test is a technique that allows the pictorial evaluation of emotional states using three parameters: pleasure, excitement, and dominance, which are associated with a person's emotional reaction. SAM is an inexpensive and easy method to evaluate affective response reports in many contexts quickly. The output used to supervise the neural network training was the result obtained through the image; the SAM test gave us a qualitative description emotion associated with the image.

Once the dataset was built, the next step was to train the model. To do this, six features were extracted from each biosignal [40], which would allow us to perform the classification. To extract the main characteristics of this database, the equations presented by Picard [39] were used. Picard defined six equations to extract biological signal characteristics using statistical methods. Using these equations, these characteristics were extracted from PPG and GRS signals. This allowed us to use these data as input for the emotion classification algorithm, which used in-depth learning as a tool to perform this classification. Our classifier was composed of a 1D CNN (1D Convolutional Neural Network), and the network structure is shown in Figure 5.

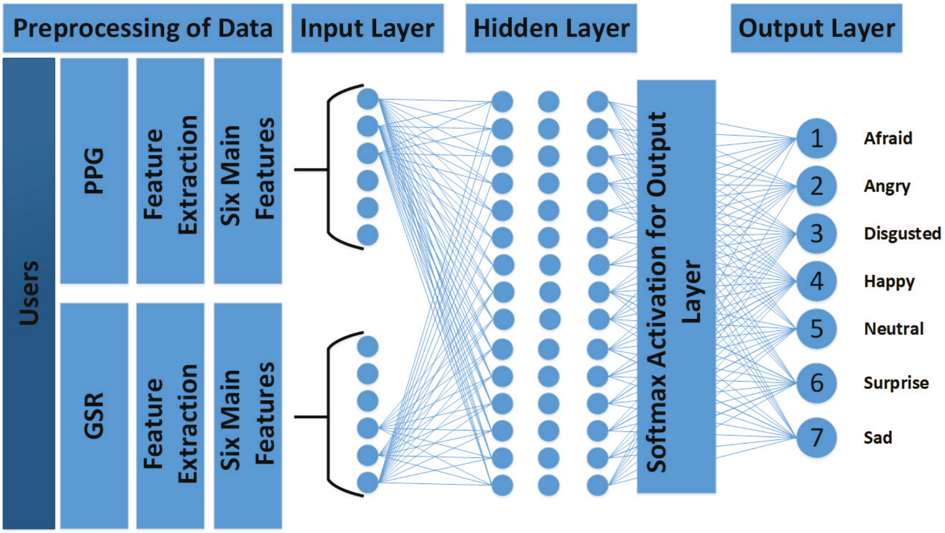


Figure 5. Structure of the 1D CNN used to classify emotions.

Figure 6 shows the accuracy between the input and validation data; likewise, you can observe the loss during the same process (training and validation).

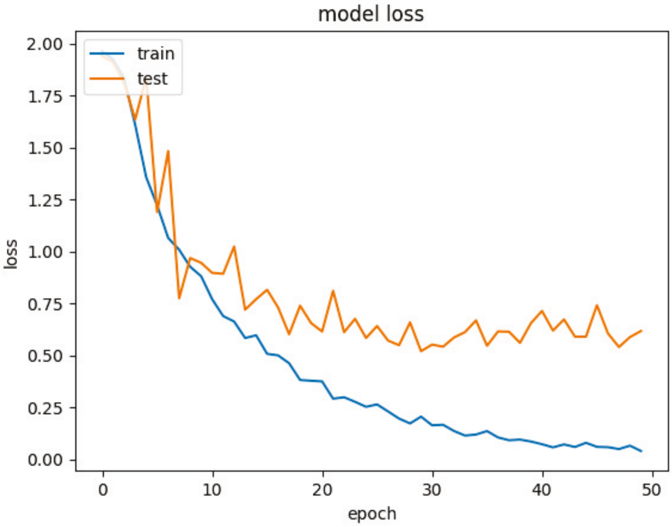


Figure 6. Model accuracy and loss of emotion recognition.

Those hyper-parameters were used in the experiments carried out in [12]. Due to the good results that were obtained there, it was decided to use the same parameters for this paper.

The network had 12 neurons in the input layer, and these corresponded to the 12 characteristics extracted from the signals (six for each signal). The hyper-parameters of the 1D-CNN are shown in Table 1.

Table 1. 1D-CNN's hyper-parameters to classify activities.

L2 Regularization or l2-penalty	0.01
Hidden Layers	[32, 64, 128, 64, 32]
Dropout Rate	0.2
Monitor	val_loss
Min. Delta	10

3.2.2. Exercise Classification

Physical exercise has a direct impact on human health. Studies have shown that frequent exercise performed by older people [41] helps to reduce the risk of: stroke or heart attack, decreased bone density, developing dementia, common diseases; and boost confidence and independence. In the vast majority of cases, these exercises require the supervision of a specialist, a physiotherapist, or an expert in sports. These experts suggest the exercises to be performed, based on age and physical limitations or injuries. In some cases, this staff has to follow up, determining whether the exercises are being performed correctly. The expert recognizes whether the exercise is being done properly or not based on experience.

We propose a device to monitor remotely, capturing the movements of the wearer through two accelerometers using low energy Bluetooth for communication. These data were sent to the smartphone, which was responsible for recognizing the activity using deep learning techniques. As there was no public database, it was decided to develop our own database. This database contained five exercises, which were carried out by people aged between 30 and 50. During the exercises, people were accompanied by a physiotherapist who was responsible for determining whether the exercise was carried out correctly. Each one of the exercises: chest stretch, arm raises, one-leg stand, bicep curls, and sideways walking, had a total of 31 participants, and a total of 1000 samples was collected per exercise.

The database contained 150,000 signals; one has to be aware that these data were tripled. This was mainly because the three axes of the accelerometer (X, Y, Z) were stored, so that in the end, we obtained a database of 450,000 signals. From this database, the following partition was made to perform the training, test, and validation of our model: training 80%, test 10%, and validation 10%.

Figure 7 shows the different steps carried out for the classification of physical activities.

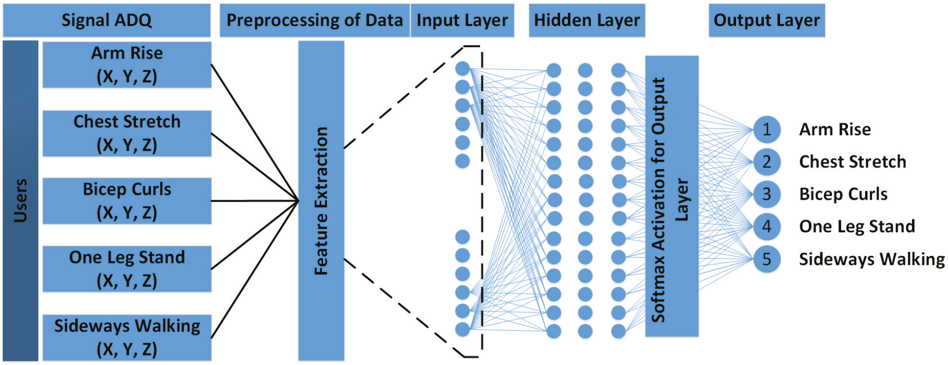


Figure 7. Structure of the 1D CNN used to classify activities.

The entries that allowed carrying out the classification of the activities was the acceleration in the three axes of each of the activities. The realized activities, as well as the captured signals for each of them can be seen in Figures 8–12 (the hyper-parameters of the 1D-CNN are shown in Table 2). Once the signals were obtained, they were reshaped, creating a 6415 × 150 matrix (Table 3). Once the matrix was reshaped, the data were sent to the neural network for classification.

Table 2. 1D-CNN's hyper-parameters to classify activities.

Layer (Type)	Output Shape	Param#
reshape_1 (Reshape)	(None, 50, 3)	0
conv1d_1 (Conv1D)	(None, 41, 100)	3100
conv1d_2 (Conv1D)	(None, 32, 100)	100,100
max_pooling1d_1 - MaxPooling1	(None, 32, 100)	0
conv1d_3 (Conv1D)	(None, 23, 180)	180,180
conv1d_4 (Conv1D)	(None, 14, 180)	324,180
global_average_pooling1d_1	(None, 180)	0
dropout_1 (Dropout)	(None, 180)	0
dense_1 (Dense)	(None, 5)	905

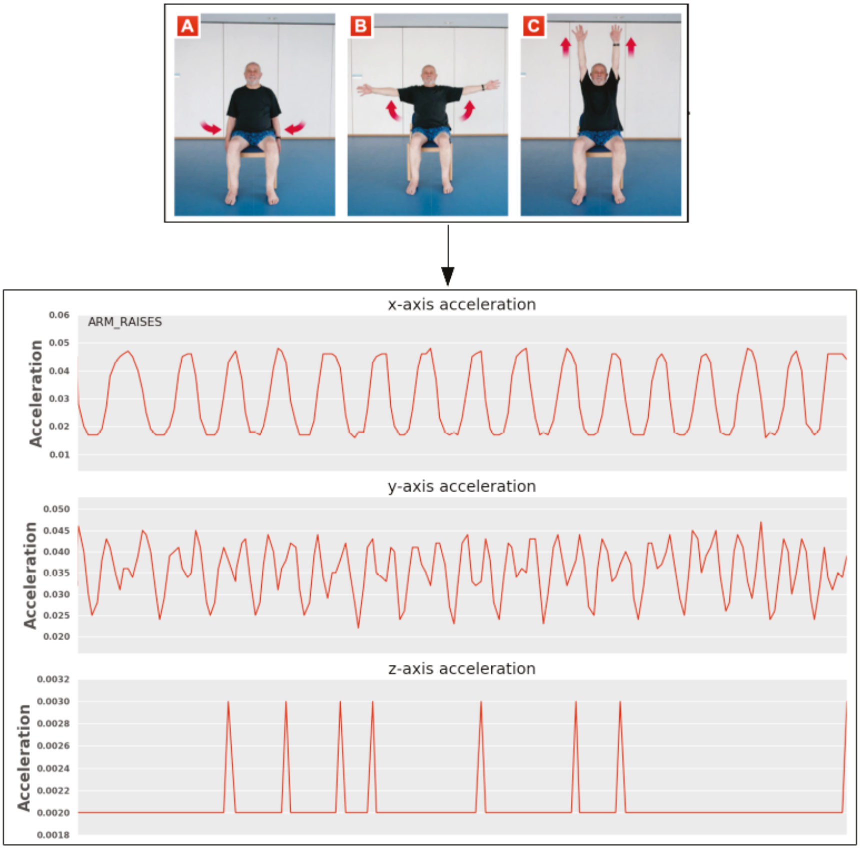


Figure 8. Arm rise.

Table 3. Input data from the network.

x_train shape	(6415, 50, 3)
N Training Samples	6415
y_train shape	(6415,)
x_train shape	(6415, 150)
input_shape	150

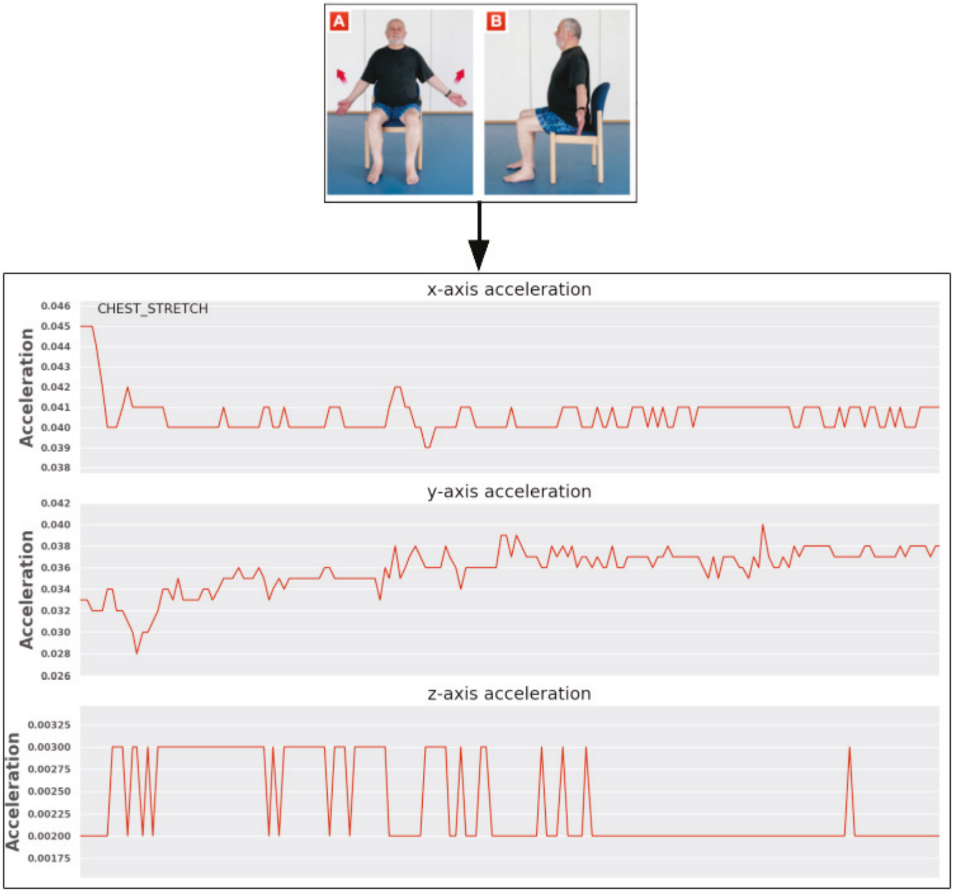


Figure 9. Chest stretch.

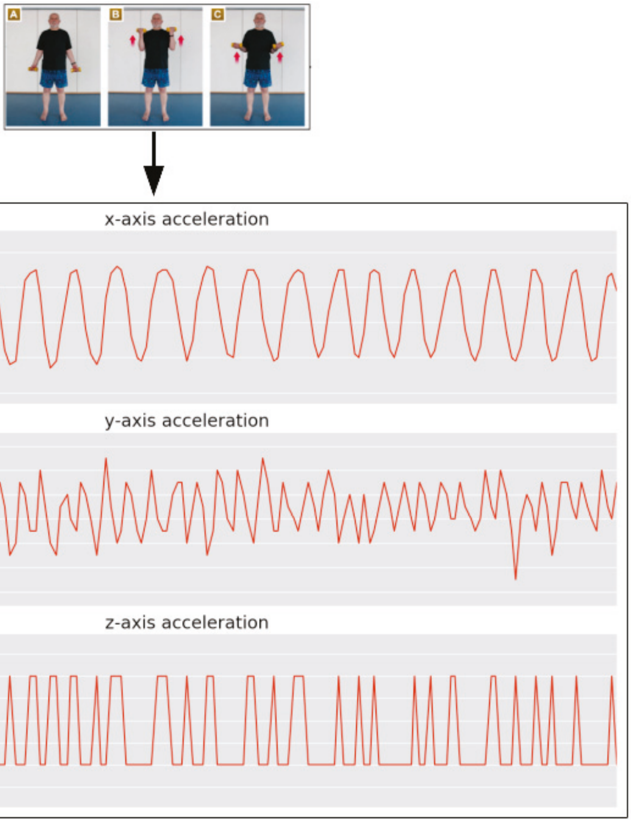


Figure 10. Bicep curls.

The results of the network are shown in Figures 13 and 14. Figure 13 shows model accuracy and loss in the training phase.

Figure 14 shows the confusion matrix, which describes the false positives and false negatives of our network. The information extracted from these graphs allowed us to determine that the model adequately recognized physical activities. This matrix showed us the number of True Positives (TP), against False Negatives (TN). Based on this matrix, we could determine that our system obtained a total of 59 TP for the first exercise, a total of 62 TP for the second, for the third exercise 62 TP, the fourth exercise 56 TP, and a total of 61 TP for the fifth exercise.

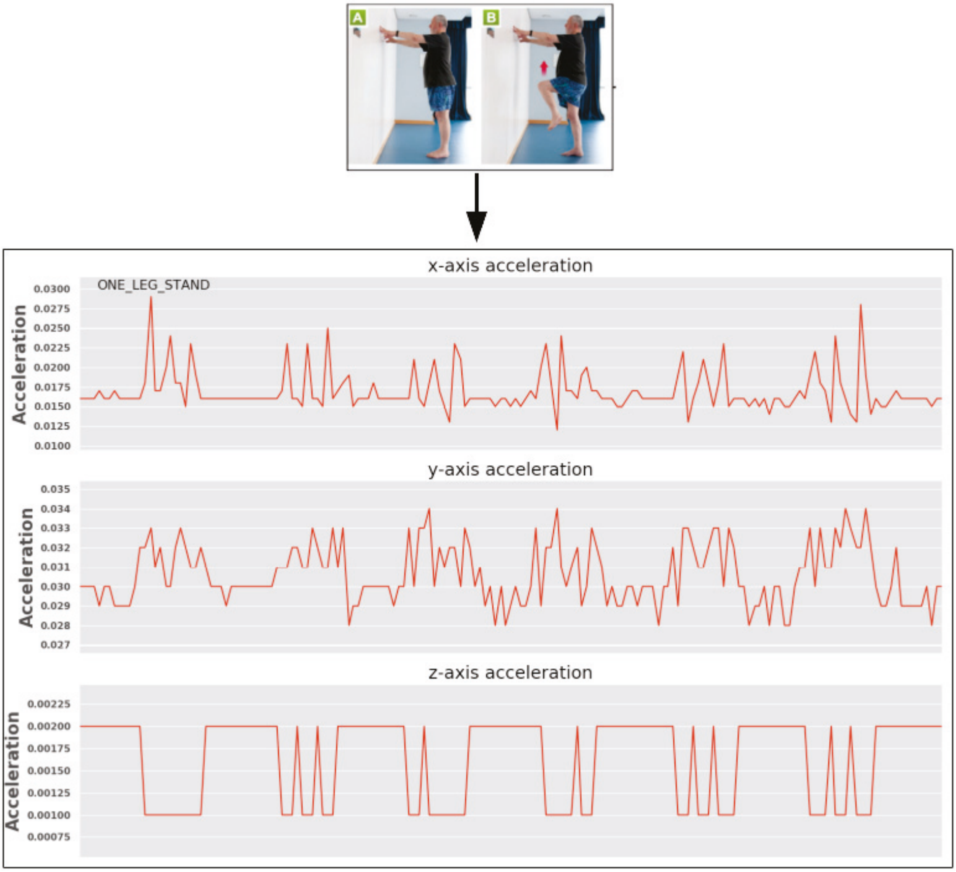


Figure 11. One leg stand.

3.2.3. Information Display

Currently, we are focused on extracting information from the data available and constructing reliable and accurate models that can be easily used in different scenarios. Nonetheless, the users and caregivers are able to visualize the saved information in a simple web-page. The objective is to build a multi-user/multi-level interface where the information is displayed in a personalized manner, showing different graphics to each type of user (the granularity of the information that should be displayed to the caregiver is very different from that that is displayed to the care receiver). Additionally, we will be using the features of a previous project, iGenda [18–20], to manage the care receivers ADL and exercises in an intelligent way, introducing cognitive help through remembering the care receivers of events and giving them advises. The aim is to improve cognition by triggering actions that help the elders to jog their memory, keeping them agile and active.

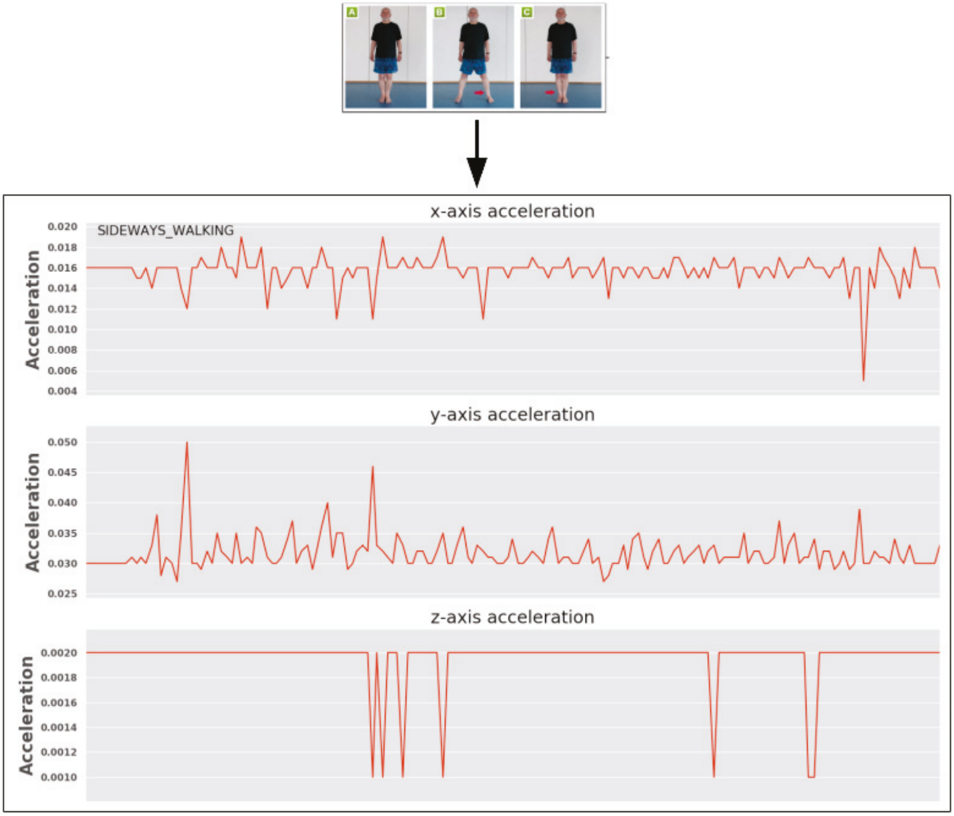


Figure 12. Sideways walking.

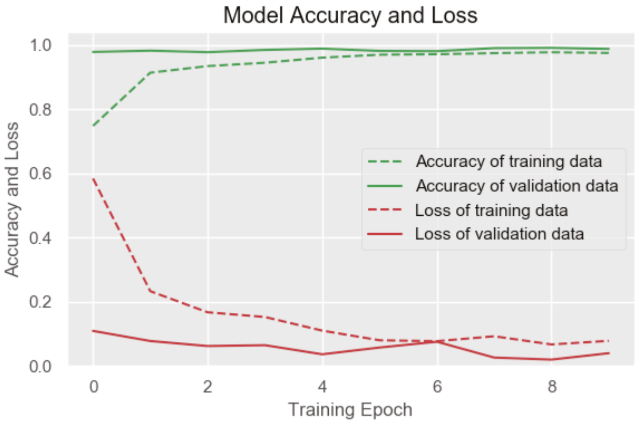


Figure 13. Model accuracy and loss.

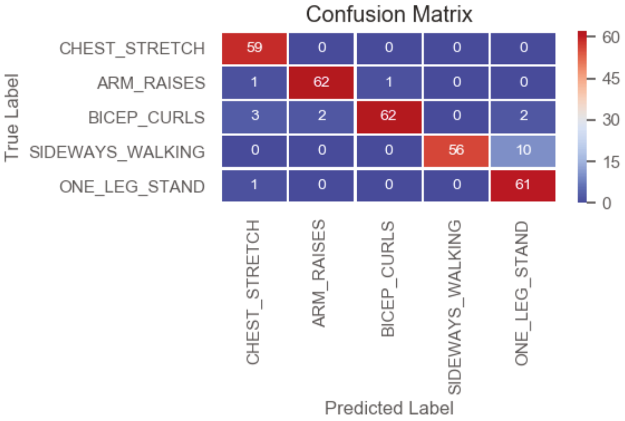


Figure 14. Confusion matrix.

4. Conclusions

This paper presented the integration of non-invasive bio-signal monitoring for the detection and classification of human emotional states and physical activities using a low-cost, easy-to-deploy sensor system. In this manner, the developed system was used to attain data for the models capable of detecting and classifying body movements and inferring the emotion that these exercises generated in patients. Therefore, it was possible to build a system that produced complex results with minimal cost.

This was an advancement of the current state-of-the-art due to the combination of several software features on just two simple low-cost devices. As stated in Section 2, there are other projects working in this domain, but they tend to focus on off-the-shelf hardware solutions, thus suffering from less-than-optimal data access of filtered data, unlike our case, as we had total control of the data. Finally, most projects use expensive solutions, which may be a barrier for most elderly people, while we used significantly less expensive solutions that may also decrease the time-to-market value.

The proposed approach was partially validated by patients and workers of a daycare center Centro Social Irmandade de São Torcato. The validation was performed through the performance of simple exercises with the patients under the supervision of caregivers. The future work will focus on the development of new tests with a higher number of users and the complete version of the visual interface. These new tests will allow us to use the information obtained to improve our learning models for a better recognition of the different activities and tasks that are performed by the patients. We will also focus our future research on determining the degree of accuracy in which the patient performs the exercise, for greater confidence in making possible corrective decisions by the caregivers.

Author Contributions: Conceptualization, A.C., J.A.R., V.J., P.N., C.C.; methodology, A.C., J.A.R., V.J., P.N., C.C.; software, A.C., J.A.R., V.J., P.N., C.C.; investigation, A.C., J.A.R., V.J., P.N., C.C.; writing, original draft preparation, A.C., J.A.R., V.J., P.N., C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Acknowledgments: This work was partly supported by the FCT (Fundação para a Ciência e Tecnología) through the Post-Doc scholarship SFRH/BPD/102696/2014 (A. Costa), by the Generalitat Valenciana (PROMETEO/2018/002), and by the Spanish Government (RTI2018-095390-B-C31).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. United Nations, Department of Economic and Social Affairs, Population Division. *World Population Prospects 2019*; Volume I: Comprehensive Tables; United Nations: New York, NY, USA, 2019.
2. Licher, S.; Darweesh, S.K.L.; Wolters, F.J.; Fani, L.; Heshmatollah, A.; Mutlu, U.; Koudstaal, P.J.; Heeringa, J.; Leening, M.J.G.; Ikram, M.K.; et al. Lifetime risk of common neurological diseases in the elderly population. *J. Neurol. Neurosurg. Psychiatry* **2018**, *90*, 148–156. [CrossRef] [PubMed]
3. Jaul, E.; Barron, J. Age-Related Diseases and Clinical and Public Health Implications for the 85 Years Old and Over Population. *Front. Public Health* **2017**, *5*, 335. [CrossRef]
4. Brasure, M.; Desai, P.; Davila, H.; Nelson, V.A.; Calvert, C.; Jutkowitz, E.; Butler, M.; Fink, H.A.; Ratner, E.; Hemmy, L.S.; et al. Physical Activity Interventions in Preventing Cognitive Decline and Alzheimer-Type Dementia. *Ann. Intern. Med.* **2017**, *168*, 30. [CrossRef]
5. Iuliano, E.; .; di Cagno, A.; Cristofano, A.; Angiolillo, A.; D'Aversa, R.; Ciccotelli, S.; Corbi, G.; Fiorilli, G.; Calcagno, G.; Costanzo, A.D. Physical exercise for prevention of dementia (EPD) study: Background, design and methods. *BMC Public Health* **2019**, *19*, 659. [CrossRef] [PubMed]
6. Müllers, P.; Taubert, M.; Müller, N.G. Physical Exercise as Personalized Medicine for Dementia Prevention? *Front. Physiol.* **2019**, *10*. [CrossRef] [PubMed]
7. Del Carmen Pérez-Fuentes, M.; Linares, J.J.G.; Fernández, M.D.R.; del Mar Molero Jurado, M. Inventory of Overburden in Alzheimer's Patient Family Caregivers with no Specialized Training. *Int. J. Clin. Health Psychol.* **2017**, *17*, 56–64. [CrossRef]
8. Berglund, E.; Lytsy, P.; Westerling, R. Health and wellbeing in informal caregivers and non-caregivers: A comparative cross-sectional study of the Swedish general population. *Health Qual. Life Outcomes* **2015**, *13*, 109. [CrossRef]
9. Peña-Longobardo, L.M.; Oliva-Moreno, J. Caregiver Burden in Alzheimer's Disease Patients in Spain. *J. Alzheimer's Dis.* **2014**, *43*, 1293–1302. [CrossRef]
10. Hoefman, R.J.; Meulenkamp, T.M.; Jong, J.D.D. Who is responsible for providing care? Investigating the role of care tasks and past experiences in a cross-sectional survey in the Netherlands. *BMC Health Serv. Res.* **2017**, *17*. [CrossRef] [PubMed]
11. Pearson, C.F.; Quinn, C.C.; Loganathan, S.; Datta, A.R.; Mace, B.B.; Grabowski, D.C. The Forgotten Middle: Many Middle-Income Seniors Will Have Insufficient Resources for Housing and Health Care. *Health Aff.* **2019**, *38*. [CrossRef]
12. Rincon, J.A.; Costa, A.; Novais, P.; Julian, V.; Carrascosa, C. Intelligent Wristbands for the Automatic Detection of Emotional States for the Elderly. In *Intelligent Data Engineering and Automated Learning—IDEAL 2018*; Springer International Publishing: Cham, Switzerland, 2018; pp. 520–530.
13. Costa, A.; Novais, P.; Julian, V.; Nalepa, G.J. Cognitive assistants. *Int. J. Hum.-Comput. Stud.* **2018**, *117*, 1–3. [CrossRef]
14. Martinez-Martin, E.; del Pobil, A.P. Personal Robot Assistants for Elderly Care: An Overview. In *Intelligent Systems Reference Library*; Springer International Publishing: Cham, Switzerland, 2017; pp. 77–91. [CrossRef]
15. Costa, A.; Martinez-Martin, E.; Cazorla, M.; Julian, V. PHAROS—PHysical Assistant RObot System. *Sensors* **2018**, *18*, 2633. [CrossRef] [PubMed]
16. Castillo, J.C.; Álvarez-Fernández, D.; Alonso-Martín, F.; Marques-Villarroya, S.; Salichs, M.A. Social Robotics in Therapy of Apraxia of Speech. *J. Healthc. Eng.* **2018**, *2018*, 1–11. [CrossRef] [PubMed]
17. CoME. 2019. Available online: http://come-aal.eu/ (accessed on 12 June 2019).
18. Costa, A.; Rincon, J.A.; Carrascosa, C.; Novais, P.; Julian, V. Activities suggestion based on emotions in AAL environments. *Artif. Intell. Med.* **2018**, *86*, 9–19. [CrossRef] [PubMed]
19. Costa, A.; Novais, P.; Simoes, R. A caregiver support platform within the scope of an ambient assisted living ecosystem. *Sensors* **2014**, *14*, 5654–5676. [CrossRef] [PubMed]
20. Costa, Â.; Heras, S.; Palanca, J.; Jordán, J.; Novais, P.; Julian, V. Using Argumentation Schemes for a Persuasive Cognitive Assistant System. In *Multi-Agent Systems and Agreement Technologies*; Springer International Publishing: Cham, Switzerland, 2017; pp. 538–546. [CrossRef]

21. Wang, J.; Chen, Y.; Hao, S.; Peng, X.; Hu, L. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognit. Lett.* **2019**, *119*, 3–11. [CrossRef]
22. Nweke, H.F.; Teh, Y.W.; Al-garadi, M.A.; Alo, U.R. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. *Expert Syst. Appl.* **2018**, *105*, 233–261. [CrossRef]
23. Martinez-Martin, E.; Cazorla, M. A Socially Assistive Robot for Elderly Exercise Promotion. *IEEE Access* **2019**, *7*, 75515–75529. [CrossRef]
24. Martinez-Martin, E.; Cazorla, M. Rehabilitation Technology: Assistance from Hospital to Home. *Comput. Intell. Neurosci.* **2019**, *2019*, 1–8. [CrossRef]
25. Cruz, E.; Escalona, F.; Bauer, Z.; Cazorla, M.; García-Rodríguez, J.; Martinez-Martin, E.; Rangel, J.C.; Gomez-Donoso, F. Geoffrey: An Automated Schedule System on a Social Robot for the Intellectually Challenged. *Comput. Intell. Neurosci.* **2018**, *2018*, 1–17. [CrossRef]
26. Vepakomma, P.; De, D.; Das, S.K.; Bhansali, S. A-Wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities. In Proceedings of the 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN), Cambridge, MA, USA, 9–12 June 2015; IEEE: Piscataway, NJ, USA, 2015. [CrossRef]
27. Cao, L.; Wang, Y.; Zhang, B.; Jin, Q.; Vasilakos, A.V. GCHAR: An efficient Group-based Context—aware human activity recognition on smartphone. *J. Parallel Distrib. Comput.* **2018**, *118*, 67–80. [CrossRef]
28. Marechal, C.; Mikołajewski, D.; Tyburek, K.; Prokopowicz, P.; Bougueroua, L.; Ancourt, C.; Wegrzyn-Wolska, K. Survey on AI-Based Multimodal Methods for Emotion Detection. In *Lecture Notes in Computer Science*; Springer International Publishing: Cham, Switzerland, 2019; pp. 307–324. [CrossRef]
29. Garcia-Garcia, J.M.; Penichet, V.M.R.; Lozano, M.D. Emotion detection. In Proceedings of the XVIII International Conference on Human Computer Interaction (Interacción'17), Cancun, Mexico, September 2017; ACM Press: New York, NY, USA, 2017.
30. Brás, S.; Ferreira, J.H.T.; Soares, S.C.; Pinho, A.J. Biometric and Emotion Identification: An ECG Compression Based Method. *Front. Psychol.* **2018**, *9*. [CrossRef] [PubMed]
31. Goshvarpour, A.; Abbasi, A.; Goshvarpour, A. An accurate emotion recognition system using ECG and GSR signals and matching pursuit method. *Biomed. J.* **2017**, *40*, 355–368. [CrossRef] [PubMed]
32. Naji, M.; Firoozabadi, M.; Azadfallah, P. A new information fusion approach for recognition of music-induced emotions. In Proceedings of the IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), Chicago, IL, USA, 19–22 May 2019; IEEE: Piscataway, NJ, USA, 2014. [CrossRef]
33. Naji, M.; Firoozabadi, M.; Azadfallah, P. Emotion classification during music listening from forehead biosignals. *Signal Image Video Process.* **2015**, *9*, 1365–1375. [CrossRef]
34. Seoane, F.; Mohino-Herranz, I.; Ferreira, J.; Alvarez, L.; Buendia, R.; Ayllón, D.; Llerena, C.; Gil-Pita, R. Wearable Biomedical Measurement Systems for Assessment of Mental Stress of Combatants in Real Time. *Sensors* **2014**, *14*, 7120–7141. [CrossRef]
35. Diaz, K.M.; Krupka, D.J.; Chang, M.J.; Peacock, J.; Ma, Y.; Goldsmith, J.; Schwartz, J.E.; Davidson, K.W. Fitbit®: An accurate and reliable device for wireless physical activity tracking. *Int. J. Cardiol.* **2015**, *185*, 138–140. [CrossRef]
36. Falter, M.; Budts, W.; Goetschalckx, K.; Cornelissen, V.; Buys, R. Accuracy of Apple Watch Measurements for Heart Rate and Energy Expenditure in Patients with Cardiovascular Disease: Cross-Sectional Study. *JMIR mHealth uHealth* **2019**, *7*, e11889. [CrossRef]
37. Rincon, J.A.; Julian, V.; Carrascosa, C.; Costa, A.; Novais, P. Detecting emotions through non-invasive wearables. *Log. J. IGPL* **2018**, *26*, 605–617. [CrossRef]
38. Porcu, S.; Uhrig, S.; Voigt-Antons, J.N.; Möller, S.; Atzori, L. Emotional Impact of Video Quality: Self-Assessment and Facial Expression Recognition. In Proceedings of the 2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX), Berlin, Germany, 5–7 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
39. Tsonos, D.; Kouroupetroglou, G. A methodology for the extraction of reader's emotional state triggered from text typography. In *Tools in Artificial Intelligence*; IntechOpen: London, UK, 2008.

40. Rincon, J.A.; Costa, A.; Carrascosa, C.; Novais, P.; Julian, V. EMERALD — Exercise Monitoring Emotional Assistant. *Sensors* **2019**, *19*, 1953. [CrossRef]
41. Kannus, P.; Sievänen, H.; Palvanen, M.; Järvinen, T.; Parkkari, J. Prevention of falls and consequent injuries in elderly people. *Lancet* **2005**, *366*, 1885–1893. [CrossRef]

electronics

MDPI

Article

Activities of Daily Living and Environment Recognition Using Mobile Devices: A Comparative Study

José M. Ferreira [1,†], Ivan Miguel Pires [2,3,*,†], Gonçalo Marques [2,†], Nuno M. Garcia [2,†], Eftim Zdravevski [4,†], Petre Lameski [4,†], Francisco Flórez-Revuelta [5,†] and Susanna Spinsante [6,†] and Lina Xu [7,†]

[1] Computer Science Department, University of Beira Interior, 6200-001 Covilha, Portugal; jose.ferreira@ubi.pt
[2] Institute of Telecommunications, University of Beira Interior, 6200-001 Covilha, Portugal; goncalosantosmarques@gmail.com (G.M.); ngarcia@di.ubi.pt (N.M.G.)
[3] Computer Science Department, Polytechnic Institute of Viseu, 3504-510 Viseu, Portugal
[4] Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, 1000 Skopje, Macedonia; eftim.zdravevski@finki.ukim.mk (E.Z.); petre.lameski@finki.ukim.mk (P.L.)
[5] Department of Computing Technology, University of Alicante, P.O. Box 99, E-03080 Alicante, Spain; francisco.florez@ua.es
[6] Department of Information Engineering, Marche Polytechnic University, 60131 Ancona, Italy; s.spinsante@staff.univpm.it
[7] School of Computer Science, University College Dublin, Dublin 4, Ireland; lina.xu@ucd.ie
* Correspondence: impires@it.ubi.pt; Tel.: +351-966-379-785
† These authors contributed equally to this work.

Received: 18 December 2019; Accepted: 14 January 2020; Published: 18 January 2020

Abstract: The recognition of Activities of Daily Living (ADL) using the sensors available in off-the-shelf mobile devices with high accuracy is significant for the development of their framework. Previously, a framework that comprehends data acquisition, data processing, data cleaning, feature extraction, data fusion, and data classification was proposed. However, the results may be improved with the implementation of other methods. Similar to the initial proposal of the framework, this paper proposes the recognition of eight ADL, e.g., walking, running, standing, going upstairs, going downstairs, driving, sleeping, and watching television, and nine environments, e.g., bar, hall, kitchen, library, street, bedroom, living room, gym, and classroom, but using the Instance Based k-nearest neighbour (IBk) and AdaBoost methods as well. The primary purpose of this paper is to find the best machine learning method for ADL and environment recognition. The results obtained show that IBk and AdaBoost reported better results, with complex data than the deep neural network methods.

Keywords: activities of daily living; AdaBoost; mobile devices; artificial neural networks; deep neural networks

1. Introduction

The use of mobile devices while doing daily activities is increasing [1]. These devices have different types of sensors that allow the acquisition of several data related to the user, including the accelerometer, magnetometer, gyroscope, Global Positioning System (GPS) receiver, and microphone [2,3]. These sensors allow the creation of intelligent systems to improve the quality of life. The monitoring of older adults or people with chronic diseases is one of the critical purposes. Furthermore, it can be useful to support sports activities and stimulate the practice of physical activity in teenagers [4]. The development of these systems is included in the research of Ambient Assisted Living (AAL) systems and Enhanced Living Environments (ELE) [5–10].

The automatic recognition of ADL is widely researched [11–16], where the previously proposed framework [2,17–25] was tested and validated with different types of Artificial Neural Networks (ANN) [26–28], verifying that the best results were achieved with Deep Neural Networks (DNN). The proposed framework allows the recognition of eight ADL, i.e., walking, running, standing, going upstairs, going downstairs, watching television, sleeping, driving, and other activities without motion, and nine environments, i.e., bar, classroom, gym, hall, kitchen, library, street, bedroom, and living room. This framework uses sensors available in mobile devices [29,30], reporting different accuracies. The proposed architecture is composed of data acquisition, data processing, data fusion, and data classification. The classification module is divided into three small stages, including the recognition of simple ADL, i.e., running, standing, walking, going upstairs, going downstairs, and other activities without motion, with accelerometer, gyroscope, and magnetometer sensors, the recognition of environments, i.e., bar, classroom, gym, hall, kitchen, library, street, bedroom, and living room, with the microphone data, and the recognition of activities without motion, i.e., sleeping, watching television, driving, and other activities without movement.

This research is based on the creation of a framework for the recognition of ADL and its environments. Still, its main goal is related to the testing of ensemble learning methods to further improve the obtained accuracy in the recognition.

The main contribution of this paper is the implementation of different machine learning methods with the same dataset used for the creation of the framework [31], including AdaBoost [32,33] and Instance Based k-nearest neighbour (IBk) [34], using different Java based frameworks, including Weka [35] and Smile [36]. Finally, the results obtained with the different methods should be compared to decide the best method for implementation using the ADL and environment recognition framework.

The results show that the application of the IBk method implemented with Weka software reported better results than others, reporting results with around 77.68% accuracy in recognition of ADL, 41.43% accuracy in recognition of environments, and 99.73% accuracy in recognition of activities without motion. However, AdaBoost applied with Smile also gave important results, reporting results between 85.44% (going upstairs) and 99.98% (driving).

Section 2 gives the presentation of the different methods implemented. The results and the comparative study of this paper are presented in Section 3. Finally, the discussion and conclusions are presented in Section 4.

2. Methods

2.1. Study Design

This study consisted of the use of the same structure and data acquired by the research presented in [18,21,22,24,25] to implement a comparative study between three types of studies. The tests were conducted with the dataset available in [24], which included data related to the eight ADL and nine environments. The information was acquired from the accelerometer, magnetometer, gyroscope, microphone, and GPS receiver available in the mobile device.

As presented in [21], an Android application was used for the acquisition of the data related to the different sensors. This mobile application is responsible for data acquisition and data processing using built-in smartphone sensors such as the accelerometer, magnetometer, gyroscope, sound, and GPS data. The software was responsible for managing five seconds of data every five minutes. It was installed in a smartphone, and it was placed in the front pocket of the pants of 25 subjects with different lifestyles, aged between 16 and 60 years old. For ADL and environment identification, a minimum of 2000 samples with five seconds of data acquired from the different sensors was available in the dataset used for this research. Different environments were used in the performed tests and were strictly related to specific activities. The volunteers had to select the ADL that would be performed using the mobile application before the start of the test. By default, the mobile application did not save any data without user input. However, the proposed method had limitations related to battery consumption and the processing power needed to perform the tests. Currently, the majority of the smartphones available on the market incorporate high performance processing units that can be used to perform the tests, and the main problem is related to power consumption. However, most people usually recharge their mobile phones daily. Therefore, the proposed method can be used in real-life scenarios.

2.2. Overview of the Framework for the Recognition of the Activities of Daily Living and Environments

Based on the previously proposed framework [20], Figure 1 shows a framework composed of four stages, including data acquisition, data processing, data fusion, and data classification. The data processing consisted of several phases, including data cleaning and feature extraction. The data classification was divided into three stages, the recognition of simple ADL (Stage 1), the identification of environments (Stage 2), and the activities without motion (Stage 3). Stage 1 included the use of the data acquired from the accelerometer, magnetometer, and gyroscope sensors. The data received from the microphone were processed in Stage 2. Finally, Stage 3 increased the number of sensors, combining the data acquired from the accelerometer, magnetometer, and gyroscope sensors with the data obtained from the GPS receiver and the environment previously recognised.

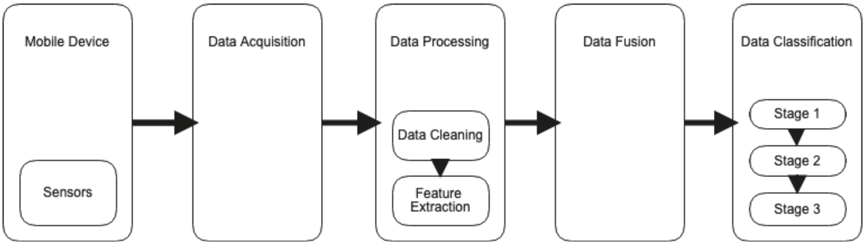


Figure 1. Flowchart of the ADL and environment recognition framework implemented in this study.

Mobile devices are composed of several sensors, which are capable of acquiring different types of data. The framework proposed was capable of acquiring and analysing 5 seconds of data and identifying the current ADL executed and the current environment frequented. The next stage consisted of the processing of the data acquired from the sensors for a further fusion of the different data acquired from the sensors. The final module of the framework consisted of the classification of the data, which started to process all features extracted from the sensors available in the mobile device and identified if the ADL executed was available in the set of ADL proposed. In the affirmative case, the ADL performed was presented to the user. Next, the environment frequented was recognised in the next stage, and it was presented to the user. If no ADL was recognised or the ADL recognized was standing, the identification of a standing ADL would be executed, trying to discover the activity performed by the user.

2.2.1. Data Acquisition

This study was based on the same dataset used in [21], which is publicly available in [31]. This dataset was composed of small sets of data (five seconds every five minutes) captured by the sensors available in the off-the-shelf mobile phones, i.e., accelerometer, magnetometer, gyroscope, microphone, and GPS receiver, and stored in the cloud. The dataset used in the presented study was created using an Android mobile application for data collection. On the one hand, the running and walking data were collected in outdoor environments. On the other hand, standing and going down and upstairs were performed inside buildings.

Moreover, the tests were conducted at different times of the day. In total, thirty-six hours of data were collected, which corresponded to 2000 samples with five seconds of raw sensor data each. Before data acquisition, the user had to use the smartphone to select the ADL that would be conducted and the time needed.

2.2.2. Data Cleaning

Data cleaning is a step performed during data processing. It is mainly used to minimise the effects of the environmental noise acquired during the acquisition of the data from the sensors. Data cleaning methods depend on the type of data acquired and the sensors used. On the one hand, a low pass filter was applied to the data obtained from the accelerometer, magnetometer, and gyroscope sensors [37]. On the other hand, the Fast Fourier Transform (FFT) [38] was used to extract the relevant information from the data collected from the microphone. There were no methods needed to clean the received data from the other types of sensors.

2.2.3. Feature Extraction

After the cleaning of the data, we extracted the features. Table 1 presents the extracted features from the selected sensors, which consisted mainly of statistical features. In Stage 1, the statistical features were mainly used, i.e., standard deviation, mean, maximum and minimum value, variance, and median, of the raw data and the peaks of the motion and magnetic sensors. It also included the calculation of the five greatest distances between calculated peaks. Stage 2 was composed of the feature acquired from the microphone, including the statistical features, i.e., standard deviation, mean, maximum and minimum value, variance, and median, of the raw data, and the calculation of 25 Mel frequency cepstrum coefficients with the microphone. Finally, Stage 3 included also the distance travelled calculated from the Global Positioning (GPS) receiver data and the environment recognised in Stage 2.

Table 1. Features extracted.

Sensor	Type of Data	Features
Accelerometer Magnetometer Gyroscope	Raw data	standard deviation, mean, maximum and minimum value, variance, and median
	Peaks	five greatest distances between peaks, mean, standard deviation, variance, and median
Microphone	Raw data	26 MFCC, standard deviation, mean, maximum value, minimum value, variance, and median
GPS receiver	Raw data	distance travelled

2.2.4. Data Fusion and Classification

Data fusion and classification were included in the last stage of the ADL and environment recognition framework. The previous studies reported that the best accuracies were achieved with the DNN method [18,21,22,24,25], and all the features are presented in Table 1. This study presents the results of the test and validation of different methods, including IBk, AdaBoost with the decision stump, and AdaBoost with the decision tree, implemented in the Java programming language for compatibility with Android based devices. The configurations used were different for the different methods implemented. Firstly, the DNN method was implemented with an activation function named sigmoid, which is a function that has the sigmoid curve, widely used as an activation function for neural networks [39]. Several learning rates were previously studied, and it was verified that we obtained better results with a value equal to 0.1. For this method, the maximum number of training iterations was established as 4×10^6. The method was implemented without distance weighting, with three hidden layers, a seed value of six, and backpropagation. The Xavier function [40] was used as an initialization function, implementing L_2 regularization [41]. Secondly, the IBk method was implemented with a batch size of 100, a *k* value of 1, and the linear nearest neighbour search algorithm [42]. Finally, in the last two methods implemented, the main difference was the weak classifier used in combination with the AdaBoost method as the decision stump classifier [43], for the first one, and the decision tree classifier [44], for the second one. Other differences were revealed, where the combination of the

AdaBoost method with the decision stump classifier was implemented with a maximum number of training iterations as 10, a seed value of 1, a batch size of 100, a weight threshold of 100, and without resampling. Thus, the combination of the AdaBoost method with the decision tree classifier was implemented with a seed value of 2, a batch size of 10, a number of maximum nodes equal to 4, and 200 as the number of trees.

Initially, we started with the identification of simple ADL, i.e., walking running, standing, going upstairs, and going downstairs, which was performed with the data acquired from the accelerometer, magnetometer, and gyroscope sensors. Secondly, the recognition of environments, i.e., bar, classroom, gym, library, street, hall, living room, kitchen, and bedroom, was performed with the data retrieved from the microphone. Finally, the recognition of activities without motion, i.e., driving, sleeping, and watching television, was performed with the data collected by the accelerometer, magnetometer, gyroscope, and GPS receiver with the inclusion of the environment recognised. Thus, the framework provided the recognition of eight ADL and nine environments.

For the implementation of the methods, the following technologies and frameworks were used:

- DNN: DeepLearning4j framework [45];
- IBk: Weka software [35];
- AdaBoost with the decision stump: Weka software [35];
- AdaBoost with the decision tree: Smile (Statistical Machine Intelligence and Learning Engine) framework [36].

3. Results

3.1. Recognition of Simple ADL

The results of simple ADL recognition with the IBk method presented around 80% accuracy using the different combinations of motion and magnetic sensors, as presented in Table 2.

Table 2. ADL recognition using the Instance Based k-nearest neighbour (IBk) method implemented with Weka software.

Sensors	Correlation Coefficient	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Root Relative Squared Error	Accuracy
Accelerometer	0.8335	0.261	0.817	21.8138%	57.7675%	73.9%
Accelerometer and Magnetometer	0.8771	0.2076	0.7011	17.2911%	49.5751%	79.23%
Accelerometer, Magnetometer, and Gyroscope	0.8781	0.2009	0.6991	16.733%	49.4287%	79.91%

AdaBoost is a binary classifier that uses a weak classier to improve the recognition of different events. The implementation of this algorithm was performed with the identification of each ADL. The results of simple ADL identification with the AdaBoost with the decision stump method implemented with Weka software are presented in Table 3, verifying that all of the ADL were recognised with an accuracy between 25.61% (going downstairs recognised with the accelerometer and magnetometer sensors) and 98.44% (standing recognised with the accelerometer, magnetometer, and gyroscope sensors).

Table 3. Accuracies of ADL recognition using the AdaBoost with the decision stump method implemented with Weka software.

ADL	Accelerometer	Accelerometer and Magnetometer	Accelerometer, Magnetometer, and Gyroscope
Going downstairs	26.24%	25.61%	37.79%
Going upstairs	31.73%	32.64%	32.91%
Running	93.13%	93.00%	92.26%
Standing	96.35%	96.58%	98.44%
Walking	37.51%	51.23%	50.87%

In addition, Table 4 presents the clarification of the values obtained in Table 3, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed by comparing the correct value with all records, we verified that the values of TP and TN were higher than others, proving the reliability of the method.

Table 4. Confusion matrix values of ADL recognition using the AdaBoost with the decision stump method implemented with Weka software (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	Accelerometer				Accelerometer and Magnetometer				Accelerometer, Magnetometer and Gyroscope			
	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN	TP
Going downstairs	7469	1061	531	939	7467	1073	533	927	7606	1017	394	983
Going upstairs	7075	630	925	1370	7379	967	621	1033	7627	1498	373	502
Running	7919	81	81	1919	7914	82	86	1918	7917	97	83	1903
Standing	7938	26	62	1974	7933	33	67	1967	7977	23	23	1977
Walking	7472	552	528	1448	7629	632	371	1368	7609	546	391	1454

Moreover, the results on the recognition of simple ADL with AdaBoost with the decision tree method implemented with the Smile framework are presented in Table 5, verifying that all of the ADL presented an accuracy between 83.79% and 99.55% using the different combinations of motion and magnetic sensors.

Additionally, Table 6 presents the clarification of the values obtained in Table 5, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed by comparing the correct value with all records, we verified that the sum of the values of TP and TN was 2000. This was the value of the number of samples equal to each activity, but the method reported a high number of FP.

Finally, the results previously obtained with the implementation of the recognition of simple ADL with the DNN method implemented with the Deeplearning4j framework are presented in Table 7, verifying that all of the ADL showed an accuracy between 66.70% and 99.35% using the different combinations of motion and magnetic sensors.

Table 5. Accuracies of ADL identification using AdaBoost with the decision tree implemented with the SMILE framework.

ADL	Accelerometer	Accelerometer and Magnetometer	Accelerometer, Magnetometer, and Gyroscope
Going downstairs	83.79%	84.21%	86.07%
Going upstairs	85.29%	84.70%	85.44%
Running	98.49%	98.47%	98.43%
Standing	99.04%	99.01%	99.55%
Walking	86.90%	89.53%	91.13%

Table 6. Confusion matrix values of ADL identification using AdaBoost with the decision tree implemented with the SMILE framework (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	Accelerometer				Accelerometer and Magnetometer				Accelerometer, Magnetometer, and Gyroscope			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
Going downstairs	1017	983	7362	638	972	1028	7449	551	974	1026	7633	367
Going upstairs	1086	914	7443	557	940	1060	7530	470	1083	917	7461	539
Running	1917	83	7932	68	1917	83	7930	70	1908	92	7935	65
Standing	1965	35	7939	61	1963	37	7938	62	1976	24	7979	21
Walking	1060	940	7620	380	1317	683	7636	364	1494	506	7619	381

Table 7. Accuracies of ADL identification using the DNN method.

ADL	Accelerometer	Accelerometer and Magnetometer	Accelerometer, Magnetometer, and Gyroscope
Going downstairs	66.70%	67.95%	77.25%
Going upstairs	84.45%	81.55%	82.40%
Running	95.45%	95.70%	95.85%
Standing	99.25%	99.20%	99.35%
Walking	86.10%	88.05%	90.09%

3.2. Recognition of Environments

The use of the IBk method for the recognition of environments using the microphone data reported an average accuracy of 41.43%, as presented in Table 8. The remaining results presented in Table 9 showed that the AdaBoost with the decision stump method implemented with Weka software had an accuracy between 10.36% and 91.78%. Next, the AdaBoost with the decision tree implemented with the SMILE framework reported an accuracy between 88.74% and 99.08%. Finally, the DNN method implemented with the Deeplearning4j framework presented an accuracy between 19.90% and 98.00%.

In addition, Table 10 presents the clarification of the values obtained in Table 9, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed by comparing the correct value with all records, we verified that the values of TP were higher in the recognition of bar, library, hall, and street. However, in the remaining classes, the values of TN were correctly recognised.

Table 8. Recognition of environments using the IBk method implemented with Weka software.

Sensors	Sound
Correlation coefficient	0.8171
Mean absolute error	0.5857
Root mean squared error	1.5574
Relative absolute error	26.3488%
Root relative squared error	60.3156%
Accuracy	41.43%

Table 9. Accuracies of recognition of environments using the AdaBoost and DNN methods.

Environments	**AdaBoost with the Decision Stump**	**AdaBoost with the Decision Tree**	**DNN**
Bar	91.78%	99.08%	22.05%
Classroom	20.67%	88.74%	37.95%
Gym	10.36%	88.87%	87.85%
Hall	40.36%	92.38%	34.80%
Kitchen	16.11%	88.89%	51.35%
Library	34.01%	91.59%	19.90%
Street	38.38%	90.92%	25.35%
Bedroom	17.88%	88.88%	98.60%
Living room	18.82%	89.20%	33.50%

Table 10. Confusion matrix values of the recognition of environments using AdaBoost with the decision stump implemented with Weka software (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	**Sound**			
	TN	**FP**	**FN**	**TP**
Bar	15,961	146	39	1854
Library	15,791	1183	209	817
Hall	15,119	645	881	1355
Kitchen	16,000	1999	0	1
Bedroom	16,000	1999	0	1
Street	15,517	1180	483	820
Classroom	16,000	1999	0	1
Living room	16,000	1999	0	1
Gym	16,000	1999	0	1

Furthermore, Table 11 presents the clarification of the values obtained in Table 5, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed comparing the correct value with all records, we verified that the values of TP were higher in the recognition of bar, library, hall, and street. However, in the remaining classes, the values of TN were also correctly recognised.

Table 11. Confusion matrix values of the recognition of environments using AdaBoost with the decision tree implemented with the SMILE framework (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	Sound			
	TP	FP	TN	FN
Bar	1917	83	15,918	82
Library	720	1280	15,767	233
Hall	1419	581	15,210	790
Kitchen	1	1999	16,000	0
Bedroom	14	1986	15,984	16
Street	787	1213	15,579	421
Classroom	148	1852	15,825	175
Living room	168	1832	15,888	112
Gym	1	1999	15,995	5

3.3. Recognition of Activities without Motion

Initially, we presented, in Table 12, the results on the recognition of activities without motion with the IBk method reporting an accuracy between 99.27% and 100% using the data acquired from the accelerometer, magnetometer, gyroscope, GPS receiver, and the environment previously identified.

Table 12. Accuracies of the recognition of activities without motion using the IBk method implemented with Weka software.

Sensors	Correlation Coefficient	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Root Relative Squared Error	Accuracy
Accelerometer and environment	1	0	0	0	0	100%
Accelerometer, Magnetometer, and Environment	1	0	0	0	0	100%
Accelerometer, Magnetometer, Gyroscope, and Environment	1	0	0	0	0	100%
Accelerometer, Distance, and Environment	0.9969	0.0042	0.0645	0.6235%	7.903%	99.58%
Accelerometer, Magnetometer, Distance, and Environment	0.9964	0.0045	0.0695	0.6734%	8.5118%	99.55%
Accelerometer, Magnetometer, Gyroscope, Distance, and Environment	0.9943	0.0073	0.0876	1.0974%	10.7201%	99.27%

Furthermore, the results of the implementation of the recognition of activities without motion with the AdaBoost with the decision stump method implemented with Weka software are presented in Tables 13 and 14, verifying that the events were recognised with an accuracy between 98.32% and 100% using the data acquired from the accelerometer, magnetometer, gyroscope, GPS receiver, and the environment previously identified.

Table 13. Accuracies of the activities' recognition without motion using the AdaBoost with the decision stump method implemented with Weka software for motion and magnetic sensors after the recognition of the environment.

	Accelerometer and Environment	**Accelerometer, Magnetometer, and Environment**	**Accelerometer, Magnetometer, Gyroscope, and Environment**
Watching television	100%	100%	100%
Sleeping	100%	100%	100%

Table 14. Accuracies of the activities' recognition without motion using the AdaBoost with the decision stump method implemented with Weka software for motion, magnetic, and location sensors after the recognition of the environment

	Accelerometer, Distance, and Environment	**Accelerometer, Magnetometer, Distance, and Environment**	**Accelerometer, Magnetometer, Gyroscope, Distance, and Environment**
Watching television	98.58%	98.98%	98.98%
Driving	100%	100%	100%
Sleeping	98.32%	98.32%	98.32%

Additionally, Tables 15 and 16 present the clarification of the values obtained in Tables 13 and 14, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed by comparing the correct value with all records, we verified that the values of TP and TN were higher than others, proving the reliability of the method.

Table 15. Confusion matrix values of the recognition of activities without motion using the AdaBoost with the decision stump method implemented with Weka software for motion and magnetic sensors after the recognition of the environment (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	**Accelerometer and Environment**				**Accelerometer, Magnetometer, and Environment**				**Accelerometer, Magnetometer, Gyroscope, and Environment**			
	TN	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**	**TP**
Watching television	2000	0	0	2000	2000	0	0	2000	2000	0	0	2000
Sleeping	2000	0	0	2000	2000	0	0	2000	2000	0	0	2000

Table 16. Confusion matrix values of the recognition of activities without motion using the AdaBoost with the decision stump method implemented with Weka software for motion, magnetic, and location sensors after the recognition of the environment (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative)

ADL	**Accelerometer, Distance, and Environment**				**Accelerometer, Magnetometer, Distance, and Environment**				**Accelerometer, Magnetometer, Gyroscope, Distance, and Environment**			
	TN	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**	**TP**
Watching television	3979	0	21	2000	3998	13	2	1987	3998	13	2	1987
Driving	4000	1	0	1999	4000	1	0	1999	4000	1	0	1999
Sleeping	3974	0	26	2000	3974	0	26	2000	3974	0	26	2000

Additionally, the results on the recognition of activities without motion with the AdaBoost with the decision tree implemented with the SMILE framework are presented in Tables 17 and 18, verifying that the events were recognised with an accuracy between 98.50% and 100% using the data acquired from the accelerometer, magnetometer, gyroscope, GPS receiver, and the environment previously identified.

Table 17. Accuracies of the activities' recognition without motion using the AdaBoost with the decision tree implemented with the SMILE framework for motion and magnetic sensors after the recognition of the environment.

	Accelerometer and Environment	**Accelerometer, Magnetometer, and Environment**	**Accelerometer, Magnetometer, Gyroscope, and Environment**
Watching television	100%	100%	100%
Sleeping	100%	100%	100%

Table 18. Accuracies of the activities' recognition without motion using the AdaBoost with the decision tree implemented with the SMILE framework for motion, magnetic, and location sensors after the recognition of the environment.

	Accelerometer, Distance, and Environment	**Accelerometer, Magnetometer, Distance, and Environment**	**Accelerometer, Magnetometer, Gyroscope, Distance, and Environment**
Watching television	99.67%	99.97%	99.97%
Driving	99.98%	99.98%	99.98%
Sleeping	99.52%	99.52%	99.50%

Tables 19 and 20 present the clarification of the values obtained in Tables 17 and 18, presenting the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. As this recognition was performed as binary recognition, i.e., the comparisons were performed comparing the correct value with all records, we verified that the values of TP and TN were higher than others, proving the reliability of the method.

Table 19. Confusion matrix values of the recognition of activities without motion using the AdaBoost with the decision tree implemented with the SMILE framework for motion and magnetic sensors after the recognition of the environment (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	Accelerometer and Environment				Accelerometer, Magnetometer, and Environment				Accelerometer, Magnetometer, Gyroscope, and Environment			
	TP	FP	TN	FN	TP	FP	TN	FN	TP	FP	TN	FN
Watching television	2000	0	2000	0	2000	0	2000	0	2000	0	2000	0
Sleeping	2000	0	2000	0	2000	0	2000	0	2000	0	2000	0

Table 20. Confusion matrix values of the recognition of activities without motion using the AdaBoost with the decision tree implemented with the SMILE framework for motion, magnetic, and location sensors after the recognition of the environment (TP = True Positive; TN = True Negative; FP = False Positive; FN = False Negative).

ADL	Accelerometer, Distance, and Environment				Accelerometer, Magnetometer, Distance, and Environment				Accelerometer, Magnetometer, Gyroscope, Distance, and Environment			
	TP	FP	TN	FN	TP	FP	TN	FN	TP	FP	TN	FN
Watching television	2000	0	3980	20	2000	0	3998	2	2000	0	3998	2
Driving	1999	1	4000	0	1999	1	4000	0	1999	1	4000	0
Sleeping	1998	2	3973	27	1998	2	3973	27	1998	2	3972	28

Finally, the results of the activity recognition without motion using the DNN method implemented with the DeepLearning4j framework are presented in Tables 21 and 22, verifying that the events were recognised with an accuracy between 79.55% and 98.50% using the data acquired from the accelerometer, magnetometer, gyroscope, GPS receiver, and the environment previously identified.

Table 21. Accuracies of the activities' recognition without motion using the DNN method for motion and magnetic sensors after the recognition of the environment.

	Accelerometer and Environment	Accelerometer, Magnetometer, and Environment	Accelerometer, Magnetometer, Gyroscope, and Environment
Watching television	94.05%	94.00%	94.15%
Sleeping	97.90%	97.85%	98.00%

Based on the results reported, Table 23 presents the average of the results obtained with the different algorithms implemented. As shown, the best results were achieved with the IBk method (99.68%) and AdaBoost with the decision tree as a weak classifier (94.05%).

The training stage was faster with IBk and AdaBoost with the decision tree than the DNN method previously implemented. These methods were less complicated to implement than the DNN method and were more efficient.

Table 22. Accuracies of the activities' recognition without motion using using the DNN method for motion, magnetic, and location sensors after the recognition of the environment.

	Accelerometer, Distance, and Environment	Accelerometer, Magnetometer, Distance, and Environment	Accelerometer, Magnetometer, Gyroscope, Distance, and Environment
Watching television	94.15%	94.25%	94.35%
Driving	80.65%	79.55%	84.15%
Sleeping	98.50%	98.30%	98.15%

Table 23. Average of the accuracy of each implemented method.

Stages	DNN	IBk	AdaBoost with the Decision Stump	AdaBoost with the Decision Tree
Stage 1	87.29%	77.68%	59.75%	91.33%
Stage 2	45.71%	41.43%	32.04%	90.95%
Stage 3	99.87%	99.73%	92.83%	99.87%
Overall	77.62%	72.95%	61.54%	94.05%

Based on the limitations of mobile devices, these methods should be implemented in the ADL and environment recognition framework to improve the results provided to the user. The results showed that the recognition of ADL and its environments was possible with the implementation of the AdaBoost, IBk, and DNN methods. It allows opportunities to create a personal digital life coach and monitor the different lifestyles. It is important for all people, because mobile devices are widely used. They exploit the possibilities to improve the quality of life.

4. Discussion and Conclusions

The implementations of DNN, IBk, AdaBoost with the decision stump, and AdaBoost with the decision tree were performed with success with the dataset previously acquired, which was based on the data received from the accelerometer, magnetometer, gyroscope, GPS receiver, and microphone. The framework was composed of data acquisition, data processing, data cleaning, feature extraction, data fusion, and data classification, to recognise eight ADL and nine environments.

In general, the overall accuracies of the methods depended on the number of sensors and resources available during data acquisition. The framework should be a function of the number of sensors available in mobile devices. The methods with an accuracy higher than 90% were the IBk method and AdaBoost with the decision tree as the weak classifier.

The AdaBoost and IBk methods reported the best results because these methods were not susceptible to overfitting in comparison with the DNN method. Notably, one of the reasons for this conclusion was the use of a weak classifier by AdaBoost that handled the discrimination of some results.

According to the previously proposed structure of a framework for the recognition of ADL and environments [2,17–25], the main focus of this study was related to the data classification module, taking into account the implementations of the other modules performed in previous studies. Previously, the DNN method was implemented, and it reported reliable results. Still, for the recognition of the environments with acoustic data, the results obtained were below the expectations, because it took many resources from the processing unit. For the validation of the different implemented methods, we performed cross-validation with 10 folds.

Following the tests of the different methods for the recognition of simple ADL, the best results were achieved with AdaBoost with the decision tree implemented with the SMILE framework, reporting an overall accuracy of 91.33% with all combinations of sensors. Still, there was a high number of FP.

In the case of the recognition of environments, the best method was also AdaBoost with the decision tree implemented with the SMILE framework, reporting an overall accuracy of 99.87%. Still, it did not recognise correctly two environments. However, the AdaBoost with the decision stump method implemented with Weka software did not recognise five environments correctly, reporting an overall accuracy of 32.04%. Finally, in the recognition of activities without motion, the results obtained with AdaBoost with the decision tree implemented with the SMILE framework were the same as the results obtained with the DNN method (99.87%).

As future work, the methods should be implemented during the development of the framework for the identification of ADL and its environments, adapting the approach to all the sensors available on mobile devices.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, writing, original draft preparation, and writing, review and editing: J.M.F., I.M.P., G.M., N.M.G., E.Z., P.L., F.F.-R., S.S., and L.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável cofinanciado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020*).

Acknowledgments: This work is funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MCTES através de fundos nacionais e quando aplicável cofinanciado por fundos comunitários no âmbito do projeto UIDB/EEA/50008/2020*). This article is based on work from COST Action IC1303 - AAPELE - Architectures, Algorithms and Protocols for Enhanced Living Environments, and COST Action CA16226 - SHELD-ON- Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information at www.cost.eu.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mobile Marketing Statistics Compilation | Smart Insights. Smart Insights, 2019. Available online: https://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/ (accessed on 11 November 2019).
2. Pires, I.; Garcia, N.; Pombo, N.; Flórez-Revuelta, F. From Data Acquisition to Data Fusion: A Comprehensive Review and a Roadmap for the Identification of Activities of Daily Living Using Mobile Devices. *Sensors* **2016**, *16*, 184. [CrossRef] [PubMed]
3. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Rodríguez, N.D. Validation Techniques for Sensor Data in Mobile Health Applications. *J. Sens.* **2016**, *2016*, 1687–725. [CrossRef]
4. Shuib, L.; Shamshirb, S.; Ismail, M.H. A review of mobile pervasive learning: Applications and issues. *Comput. Hum. Behav.* **2015**, *46*, 239–244. [CrossRef]
5. Garcia, N.M.; Rodrigues, J.J.P. (Eds.). *Ambient Assisted Living*; CRC Press: Boca Raton, FL, USA, 2015.
6. Garcia, N.M. Roadmap to the Design of a Personal Digital Life Coach. In *International Conference on ICT Innovations*; Springer: Cham, Switzerland, 2015; pp. 21–27.
7. Sousa, P.S.; Sabugueiro, D.; Felizardo, V.; Couto, R.; Pires, I.; Garcia, N.M. mHealth sensors and applications for personal aid. In *Mobile Health*; Springer: Cham, Switzerland, 2015; pp. 265–281.
8. Dobre, C.; Mavromoustakis, C.X.; Garcia, N.M.; Mastorakis, G.; Goleva, R.I. Introduction to the AAL and ELE Systems. In *Ambient Assisted Living and Enhanced Living Environments*; Butterworth-Heinemann: Oxford, UK, 2017; pp. 1–16.
9. Felizardo, V.; Sousa, P.; Sabugueiro, D.; Alexre, C.; Couto, R.; Garcia, N.; Pires, I. E-Health: Current status and future trends. In *Handbook of Research on Democratic Strategies and Citizen-Centered E-Government Services*; IGI Global: Hershey, PA, USA, 2015; pp. 302–326.
10. Goleva, R.I.; Garcia, N.M.; Mavromoustakis, C.X.; Dobre, C.; Mastorakis, G.; Stainov, R.; Trajkovik, V. AAL and ELE Platform Architecture. In *Ambient Assisted Living and Enhanced Living Environments*; Butterworth-Heinemann: Oxford, UK, 2017; pp. 171–209.
11. Banos, O.; Damas, M.; Pomares, H.; Rojas, I. On the use of sensor fusion to reduce the impact of rotational and additive noise in human activity recognition. *Sensors* **2012**, *12*, 8039–8054. [CrossRef]

12. Akhoundi, M.A.A.; Valavi, E. Multi-Sensor Fuzzy Data Fusion Using Sensors with Different Characteristics. *arXiv* **2010**, arXiv:1010.6096.
13. Paul, P.; George, T. An Effective Approach for Human Activity Recognition on Smartphone. In Proceedings of the 2015 IEEE International Conference on Engineering and Technology (Icetech), Coimbatore, India, 25 January 2015; pp. 45–47. [CrossRef]
14. Hsu, Y.-W.; Chen, K.-H.; Yang, J.-J.; Jaw, F.-S. Smartphone based fall detection algorithm using feature extraction. In Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15 October 2016; pp. 1535–1540.
15. Dernbach, S.; Das, B.; Krishnan, N.C.; Thomas, B.L.; Cook, D.J. Simple and Complex Activity Recognition through Smart Phones. In Proceedings of the 2012 8th International Conference on Intelligent Environments (IE), Guanajuato, Mexico, 14 January 2012; pp. 214–221.
16. Shen, C.; Chen, Y.F.; Yang, G.S. On Motion-Sensor Behavior Analysis for Human-Activity Recognition via Smartphones. In Proceedings of the 2016 IEEE International Conference on Identity, Security and Behavior Analysis (Isba), Sendai, Japan, 22 January 2016; pp. 1–6.
17. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Identification of Activities of Daily Living Using Sensors Available in off-the-shelf Mobile Devices: Research and Hypothesis. In *International Symposium on Ambient Intelligence*; Springer: Cham, Switzerland, 2016; pp. 121–130.
18. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S. Pattern recognition techniques for the identification of Activities of Daily Living using mobile device accelerometer. *arXiv* **2017**, arXiv:1711.00096.
19. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Goleva, R.; Zdravevski, E. Recognition of activities of daily living based on environmental analyses using audio fingerprinting techniques: A systematic review. *Sensors* **2018**, *18*, 160. [CrossRef]
20. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S. Approach for the development of a framework for the identification of activities of daily living using sensors in mobile devices. *Sensors* **2018**, *18*, 640. [CrossRef]
21. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C. Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices. In *Pervasive and Mobile Computing*; Elsevier: Amsterdam, The Netherlands, 2018; Volume 47, pp. 78–93.
22. Pires, I.M.; Teixeira, M.C.; Pombo, N.; Garcia, N.M.; ; Flórez-Revuelta, F.; Spinsante, S.; Goleva, R.; Zdravevski, E. Android Library for Recognition of Activities of Daily Living: Implementation Considerations, Challenges, and Solutions. *Open Bioinform. J.* **2018**. [CrossRef]
23. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Framework for the Recognition of Activities of Daily Living and their Environments in the Development of a Personal Digital Life Coach. *DATA* **2018**. [CrossRef]
24. Pires, I.M.S. Multi-Sensor Data Fusion in Mobile Devices for the Identification of Activities of Daily Living. Ph.D. Thesis, Universidade da Beira Interior, Covilhã, Portugal, November 2018.
25. Pires, I.M.; Marques, G.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C.; Zdravevski, E. Recognition of Activities of Daily Living and Environments Using Acoustic Sensors Embedded on Mobile Devices. *Electronics* **2019**, *8*, 1499. [CrossRef]
26. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]
27. Costarelli, D.; Vinti, G. Pointwise and uniform approximation by multivariate neural network operators of the max-product type. *Neural Netw.* **2016**, *81*, 81–90. [CrossRef] [PubMed]
28. Gripenberg, G. Approximation by neural networks with a bounded number of nodes at each level. *J. Approx. Theory* **2003**, *122*, 260–266. [CrossRef]
29. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Limitations of the Use of Mobile Devices and Smart Environments for the Monitoring of Ageing People. In Proceedings of the 4th International Conference on Information and Communication Technologies for Ageing Well and e-Health, Madeira, Portugal, 22–23 March 2018; pp. 269–275.
30. Pires, I.; Felizardo, V.; Pombo, N.; Garcia, N.M. Limitations of energy expenditure calculation based on a mobile phone accelerometer. In Proceedings of the 2017 International Conference on High Performance Computing & Simulation (HPCS), Genoa, Italy, 17–21 July 2017.

31. August 2017—Multi-Sensor Data Fusion in Mobile Devices for the Identification of Activities of Daily Living. Available online: https://github.com/impires/August_2017-_Multi-sensor_data_fusion_in_mobile_devices_for_the_identification_of_activities_of_dail (accessed on 20 February 2019).
32. Yoav, F.; Schapire, R.E. A Decision-Theoretic Generalisation of on-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1995**, *55*, 119.
33. Hastie, T.; Rosset, S.; Zhu, J.; Zou, H. Multi-class AdaBoost, 2009. *Stat. Interface* **2008**, *2*, 349–360. [CrossRef]
34. Pollettini, J.T.; Panico, S.R.; Daneluzzi, J.C.; Tinós, R.; Baranauskas, J.A.; Macedo, A.A. Using machine learning classifiers to assist healthcare-related decisions: Classification of electronic patient records. *J. Med. Syst.* **2012**, *36*, 3861–3874. [CrossRef]
35. Frank, E.; Hall, M.; Reutemann, P.; Trigg, L. Weka 3—Data Mining with Open Source Machine Learning Software in Java, 2019. Available online: https://www.cs.waikato.ac.nz/ml/Weka/index.html (accessed on 10 November 2019).
36. Github, Smile—Statistical Machine Intelligence and Learning Engine, 2019. Available online: http://haifengl.github.io/smile/ (accessed on 10 November 2019).
37. Graizer, V. Effect of low-pass filtering and re-sampling on spectral and peak ground acceleration in strong-motion records. In Proceedings of the 15th World Conference of Earthquake Engineering, Lisbon, Portugal, 28 September 2012; pp. 24–28.
38. Rader, C.; Brenner, N. A new principle for fast Fourier transformation. *IEEE Trans. Acoust. Speech Signal Process.* **1976**, *24*, 264–266. [CrossRef]
39. Karlik, B.; Olgac, A.V. Performance analysis of various activation functions in generalized MLP architectures of neural networks. *Int. J. Artif. Intell. Expert Syst.* **2011**, *1*, 111–122.
40. Kumar, S.K. On weight initialization in deep neural networks. *arXiv* **2017**, arXiv:1704.08863, 2017.
41. Van Laarhoven, T. L2 regularization versus batch and weight normalization. *arXiv* **2017**, arXiv:1706.05350.
42. Nene, S.A.; Nayar, S.K. A simple algorithm for nearest neighbor search in high dimensions. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 989–1003. [CrossRef]
43. Kawaguchi, S.; Nishii, R. Hyperspectral image classification by bootstrap AdaBoost with random decision stumps. *IEEE Trans. Geosci. Remote. Sens.* **2007**, *45*, 3845–3851. [CrossRef]
44. Safavian, S.R.; Lgrebe, D. A survey of decision tree classifier methodology. *IEEE Trans. Syst. Man Cybern.* **1991**, *21*, 660–674. [CrossRef]
45. Nicholson, A.C. Deeplearning4j: Open-source, Distributed Deep Learning for the JVM, 2 Sepember 2017. Available online: https://deeplearning4j.org/ (accessed on 10 November 2019).

electronics

Article

Recognition of Activities of Daily Living and Environments Using Acoustic Sensors Embedded on Mobile Devices

Ivan Miguel Pires [1,2,*,†], Gonçalo Marques [1,†], Nuno M. Garcia [1,†], Nuno Pombo [1,†], Francisco Flórez-Revuelta [3,†], Susanna Spinsante [4,†], Maria Canavarro Teixeira [5,6,†] and Eftim Zdravevski [7,†]

[1] Instituto de Telecomunicações, Universidade da Beira Interior, 6200-001 Covilhã, Portugal; goncalosantosmarques@gmail.com (G.M.); ngarcia@di.ubi.pt (N.M.G.); ngpombo@di.ubi.pt (N.P.)

[2] Computer Science Department, Polytechnic Institute of Viseu, 3504-510 Viseu, Portugal

[3] Department of Computing Technology, University of Alicante, P.O. Box 99, E-03080 Alicante, Spain; francisco.florez@ua.es

[4] Department of Information Engineering, Università Politecnica delle Marche, 60131 Ancona, Italy; s.spinsante@staff.univpm.it

[5] UTC de Recursos Naturais e Desenvolvimento Sustentável, Polytechnique Institute of Castelo Branco, 6001-909 Castelo Branco, Portugal; ccanavarro@ipcb.pt

[6] CERNAS-Research Centre for Natural Resources, Environment and Society, Polytechnique Institute of Castelo Branco, 6001-909 Castelo Branco, Portugal

[7] Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, 1000 Skopje, Macedonia; eftim.zdravevski@finki.ukim.mk

* Correspondence: impires@it.ubi.pt; Tel.: +351-966-379-785

† These authors contributed equally to this work.

Received: 23 November 2019; Accepted: 3 December 2019; Published: 7 December 2019

Abstract: The identification of Activities of Daily Living (ADL) is intrinsic with the user's environment recognition. This detection can be executed through standard sensors present in every-day mobile devices. On the one hand, the main proposal is to recognize users' environment and standing activities. On the other hand, these features are included in a framework for the ADL and environment identification. Therefore, this paper is divided into two parts—firstly, acoustic sensors are used for the collection of data towards the recognition of the environment and, secondly, the information of the environment recognized is fused with the information gathered by motion and magnetic sensors. The environment and ADL recognition are performed by pattern recognition techniques that aim for the development of a system, including data collection, processing, fusion and classification procedures. These classification techniques include distinctive types of Artificial Neural Networks (ANN), analyzing various implementations of ANN and choosing the most suitable for further inclusion in the following different stages of the developed system. The results present 85.89% accuracy using Deep Neural Networks (DNN) with normalized data for the ADL recognition and 86.50% accuracy using Feedforward Neural Networks (FNN) with non-normalized data for environment recognition. Furthermore, the tests conducted present 100% accuracy for standing activities recognition using DNN with normalized data, which is the most suited for the intended purpose.

Keywords: Activities of Daily Living (ADL); data fusion; environments; feature extraction; pattern recognition; sensors

1. Introduction

Data collection [1] can be conducted using different sensors existing on mobile devices, such as the microphone, the accelerometer, the magnetometer and the gyroscope. The acquired data from mobile sensors are related to the movement and environment where the activities are performed [2]. These data can also be used to develop a method for automatic Activities of Daily Living (ADL) and environment recognition [3].

In continuation of a previous study, available in Reference [4], this paper proposes the use of the microphone for environment identification, that is, bar, classroom, gym, street, kitchen, hall, living room, library and bedroom, which is fused with the data collected using the accelerometer, gyroscope and magnetometer sensors for the recognition of the standing activities, that is, sleeping and watching TV. These methods are included in the design of an ADL and environment recognition framework, proposed in References [5–7]. The advantages of environment recognition are not limited to the increasing number of ADL recognized. Furthermore, this allows the framework to combine the environments with ADL recognition, which returns different results, such as the user walking on the street.

The topic related to the recognition of the ADL has some studies available in the literature [8–13] but there are no studies that use all sensors incorporated in the mobile devices. However, the Artificial Neural Network (ANN) is one of the most used methods in this topic [14,15]. Based on our previous studies using motion and magnetic sensors for the development of an environment and ADL recognition framework [4,16], this paper proposes the creation of several methods to adapt the framework to all sensors incorporated in mobile devices. Some methods using different combinations of sensors are presented in previous studies [4,16], such as the accelerometer, using the accelerometer and magnetometer and using all of the previously described, along with the gyroscope. Thus, this study presents an approach using acoustic data for environment identification, as well as different methods, fusing the environment recognized with other data sources. The proposed method can use the accelerometer and the environment, the accelerometer, the magnetometer and environment but also can be performed using all the mobile sensors and the environment (accelerometer, magnetometer and gyroscope). For the implementation and testing of these methods, we propose the use of ANN [17–19] using three different implementations of ANN [4]. This research also includes the definition of the correct set of features needed and the best implementation of ANN for ADL and environment recognition. The best results are achieved with Feedforward Neural Network (FNN) with Backpropagation for environment recognition and with Deep Learning techniques for standing activities identification.

The main goal of this study is the design of an ADL and environment recognition framework. We discovered that the recognition of the environment increases the number of activities recognized, differentiating the standing activities, where the proposed standing activities are sleeping and watching TV. At this point, the framework will be able to recognize six activities and nine environments, utilizing the accelerometer, gyroscope, magnetometer and mobile microphone sensors.

The Introduction section is concluded in this paragraph and the remaining sections are structured as follows—Section 2 introduces a literature review focused on the use of acoustic sensors for ADL and environment recognition. The methods used for the development of the ADL and environment recognition framework are presented in Section 3. Section 4 presents the results of the implementation of different methods. Finally, the discussion about the results and implementation in the framework is presented in Section 5, the conclusions are presented in Section 6.

2. Related Work

There are no studies related to the use of the fusion of the data collected using all sensors incorporated in off-the-shelf portable devices, including accelerometer, gyroscope, magnetometer and microphone, for ADL and environment recognition [1]. However, numerous methods which incorporate subsets of these mobile sensors are presented in the literature.

The authors of Reference [20] used the Global Positioning System (GPS), accelerometer and microphone sensors for sleeping, walking, standing, running, and social interaction activities recognition using linear and logistic regression methods reporting an accuracy around 90%.

In Reference [21], the authors extracted the minimum, difference between axis, mean, standard deviation, variance, correlation between axis, sum of coefficients, spectral energy and spectral entropy from the accelerometer sensor. Moreover, they study the total spectrum power, zero-crossing rate, spectral centroid, sub-band powers, spectral spread, spectral roll-off, spectral flux and Mel-Frequency Cepstral Coefficients (MFCC) using the microphone. The proposed study applied Gradient Boosting Decision Tree methods and Support Vector Machine (SVM) to recognize several activities such as sitting on a chair, standing, lying, walking, going upstairs and downstairs, running, jogging and drinking. The results report 89.12% and 91.5% accuracy.

The authors of Reference [22] recognized several activities, including cycling, cleaning table, shopping, travelling by car, going to the toilet, cooking, watching television, eating, driving, working at a computer, reading and sleeping, using data acquired from the microphone and accelerometer sensors and applying the Gaussian mixture model (GMM) with log power and MFCC as features, reporting an accuracy of 77.9%.

In Reference [23], the accelerometer and microphone sensors were also used for the recognition of shopping, driving, travelling by car, cooking, washing dishes, cleaning with a vacuum cleaner, waiting in a queue, sleeping, working at a computer, watching television, sitting, being a bar, walking, lying and standing activities, using a J48 decision tree, logistic model tree (LMT) and functional tree (FT), and Instance-based k-Nearest Neighbour (IBk) lazy algorithm with mean, standard deviation, angular degree, range and MFCC as features. The reported accuracies are around 90%, where the LMT decision tree reports 90.4%, the J48 decision tree reports 90.7%, the IBk lazy algorithm reports 90.8% and the FT decision tree reports 90.7% [23].

The remaining studies available in the literature using acoustic sensors do not use data fusion techniques, because they only use the microphone signal. Based on the acoustic signal acquired from the microphone, the authors of Reference [24] used the SVM method with spectral roll-off, slope, minimum, median, coefficient of variation, inverse coefficient of variation, trimmed mean, skewness, kurtosis and 1st, 57th, 95th and 99th percentiles as features. This method presents an accuracy higher than 90% for the recognition of some environments such as restaurant, casino, playground, train, street with ambulance, street traffic, nature at day, nature at night, river and ocean.

In Reference [25], the Linear Discriminant Classifier (LDC) was used with microphone data to recognize several ADLs, including eating, drinking, clearing the throat, relaxing, laughing, coughing, sniffling and talking. This method uses several features including log power, total Root-Mean-Square (RMS) energy, spectral kurtosis, spectral centroid, spectral roll-off, spectral flux, spectral skewness, spectral slope, spectral variance, MFCC, zero crossing rate, minimum, mean, median, maximum, RMS, 1st and 3rd quartiles, interquartile range, standard deviation, skewness, kurtosis, quantity of peaks, mean peaks distance, mean peaks amplitude, mean crossing rate and linear regression slope. The best reported accuracy was achieved using the total RMS energy, spectral flux, spectral centroid, spectral skewness, spectral variance, spectral roll-off, spectral kurtosis, spectral slope and MFCC as features. The average of the reported accuracy was 66.5%.

Artificial Neural Networks (ANN) is one of the most used methods for ADL and environment identification using acoustic signals. In Reference [26], the authors implemented an ANN method, *i.e.*,(Multilayer Perceptron) MLP, with MFCC as features for the identification of acoustic warning signals of emergency units (police, fire department and ambulance), reporting a highest accuracy of 96.7%.

Another study [27] uses ANN for the recognition of several materials collisions such as boll, metal, wood and plastic. Moreover, this research also focuses on the identification of other activities such as door opening/closing, typewriting, knocking, a phone ringing, grains falling, spray and whistle, using time-variance and frequency-variance patterns as features, reporting an average accuracy of 98%.

In Reference [28], the ANN was used for the recognition of sneezing, dog barking, clock ticking, baby crying, crowing rooster, raining, sound of sea waves, fire crackling, sound of helicopter and sound of chainsaw with some features, such as zero crossing rate, MFCC, spectral flatness and spectral centroid, reporting an accuracy around 94.5%.

The authors of Reference [29] used the FNN for the recognition of the sound of sirens from emergency vehicles, automobile horns and normal street sounds with MFCC and zero crossing rate as features, reporting an accuracy between 80% and 100%.

Deep Neural Network (DNN) is another type of ANN used for laughing, singing, crying, arguing and sighing recognition with MFCC as features [30]. The authors of Reference [31] also used DNN for the ambient scene analysis (i.e., voice, music, water and traffic), stress, emotion and speaker recognition with MFCC as features, presenting an accuracy between 60% and 90%.

The SVM is another method used for ADL and environment recognition using acoustic signals. In Reference [32], the authors achieved an accuracy of 78.4% by using the SVM method for keystrokes identification with MFCC as features. Furthermore, the SVM method has been used by the authors of Reference [33] for the identification of several sounds, including beach, forest, street, shaver, crowd football, birds, dog, sink, dishwasher, washing machine, brushing teeth, speech, bus, car, restaurant, phone ringing, train station, chair, vacuum cleaner, coffee machine, raining and computer keyboard, using MFCC as features and reporting an accuracy around 80%. The SVM method is also used for the recognition of sleeping using MFCC and sound pressure level (SPL) as features, reporting accuracies between 75% and 81% [34,35].

The Hidden Markov model (HMM) is another method used for ADL and environment recognition using acoustic signals. In Reference [36], the authors used HMM for the recognition of several sounds such as automobile, aircraft, moped, train and truck. The proposed study has used calculation and storage of sound levels, statistical indices, one-third-octave spectra and noise events detection based on thresholds as features, presenting more than 95% accuracy. In Reference [37], the authors recognized the idle state and the cicada singing sounds with HMM, based on the frequency bands and ratio.

The Gaussian Mixture Model (GMM) is another method used for ADL and environment recognition using acoustic signals. In Reference [38], the authors used GMM with MFCC as features for the recognition of calls during driving, reporting an accuracy around 86%. On the other hand, the authors of Reference [39] used GMM with zero crossing rate, Root Mean Square (RMS), MFCC and low energy frame rate as features for the recognition of emotional states, reporting an accuracy between 65% and 100%.

The authors of Reference [40] used Random Forests and SVM methods for the recognition of street music, siren, gun shot, idling, drilling, dog bark, children playing, car horn and air conditioner sounds. This study used MFCC and motif features, reporting an accuracy between 26.45% and 55.68% with SVM, and between 70.55% and 85% with Random Forests.

In Reference [41], the authors used the decision tree and HMM approach for several ADL and environment identification including reading, meeting, chatting, assisting conference talks, lectures, music, driving, elevator, walking, airplane, fan, vacuuming, shower, clapping, raining, climbing stairs, and wind. The proposed method used a zero crossing rate, low energy frame rate, spectral roll-off, spectral flux, bandwidth, normalized weighted phase deviation, and Relative Spectral Entropy (RSE). The reported accuracy is higher than 78%.

The authors of Reference [42] implemented the GMM, Feed-Forward DNN, Recurrent Neural Networks (RNN), and SVM for the recognition of baby crying and smoking alarm, using MFCC, spectral centroid, spectral flatness, spectral roll-off, spectral kurtosis and zero crossing rate, reporting accuracies between 2% and 24%.

The SVM, diverse density (DD) and expected maximization (EM) methods were implemented in Reference [43] for the recognition of several sounds, including cutlery, water, voice, ambient and music. The proposed method uses MFCC, spectral flux, spectral centroid, bandwidth, Normalized

Mel-Frequency Bands, zero crossing rate and low energy frame rate as features, presenting 87% accuracy (average).

In Reference [44], several sounds were identified, including coffee machine brewing, hand washing, walking, elevator, door opening/closing and silence, using k-Nearest Neighbour (k-NN), SVM and GMM methods. This study use several features, such as zero crossing rate, short-time energy, temporal centroid, energy entropy, autocorrelation, RMS, spectral centroid, spectral roll-off point, spectral spread, spectral entropy, spectral flux, and MFCC methods. The highest accuracies achieved with the different methods are 97.9%, with k-NN, 90%, with GMM, and 100% with SVM [44].

The authors of Reference [45] implemented the Random Forests, HMM, GMM, SVM, ANN, k-NN, and deep belief network methods to recognize babble, driving, machinery, crowded restaurant, street, air conditioner, washer, dryer, and vacuum cleaner, with MFCC, band periodicity and band entropy.

In Reference [46], the authors implemented Naive Bayes, k-NN, Random Forests and Bayesian Networks methods for the recognition of several nursing activities, including the measurement of height, patient sitting, assisting doctor, attaching/measuring/removing electrocardiography (ECG), changing bandage, cleaning body, examining edema and washing hands. This method uses several features, including mean of intensity, mean, variance of intensity, variance, mean of Fast Fourier Transform (FFT)-domain energy, and covariance between intensities. The results reported are 56.10%, with k-NN and Naive Bayes, 73.18%, with k-NN and Bayesian Networks, 55.15%, with Naive Bayes only, 80.96%, with Naive Bayes and Bayesian Networks, 59.03%, with Random Forests and Naive Bayes, and 67.83%, with Random Forests and Bayesian Networks [46].

The identification of various sounds including alarms, birds, clapping, dogs, footsteps, motorcycles, raining, rivers, sea waves, and wind, using k-NN, Naive Bayes, SVM, C4.5 decision tree, logistic regression and ANN, imputing several features is proposed in Reference [47]. These features include skewness, zero crossing rate, kurtosis, spectral spread, spectral roll-off, spectral centroid, spectral flatness measure, spectral slope, spectral flux, spectral skewness, spectral kurtosis, spectral sharpness, spectral crest factor, spectral smoothness, spectral variability, Chroma vectors and MFCC. The highest reported accuracies are 45%, with k-NN, 45%, with Naive Bayes, 54%, with SVM, 45%, with a C4.5 decision tree, 44%, with logistic regression and 54%, with ANN [47].

In Reference [48], a fall detection method was developed with k-NN, SVM, least squares method (LSM), and ANN methods with spectrogram, MFCC, linear predictive coding (LPC) and matching pursuit (MP) as features, reporting 98% accuracy.

The Random Forests classifier was also implemented for the recognition of babble, driving, go to the supermarket, outdoor walking, multiple speakers and kitchen hood. This method use band-periodicity, bandentropy, spectrum flux (SF), subband short-time energy deviation (STED) and subband power spectral deviation (SPSD) as features extracted from the microphone, and present more than 70% accuracy [49]. In Reference [50], the Random Forest was also used to recognize several activities, including using an escalator, an elevator, a drink vending machine and a ticket vending machine, crossing a gate, climingb straight stairs, waiting, entering, queuing, and getting off a train. This study implemented several features extracted from the microphone, such as the step interval, the average step interval variances, the trajectory stretchiness, the peak and trough strength and the amplitude.

The cough sound was recently recognized with a microphone, implementing the k-NN with Hu moment as features [51], which reports accuracies over 93%. Moreover, the the k-NN and the SVM methods are implemented with MFCC, Spectral Centroid, Spectral Bandwidth, Spectral Crest Factor, Spectral Turbulence, Spectral Flux, Ratio f50 versus f90, Spectral Roll-off, Spectral Standard Deviation, Spectral Skewness, Spectral Kurtosis, Spectral Peak Entropy and Tsallis Entropy as features [52], which has accuracies around 99%.

The HMM was also used with the microphone and accelerometer incorporated in mobile and wearable devices for the recognition of different scenes, including meal, arm gestures of eating, conversations, participants, TV viewing, clattering sound, and voice. This study used MFCC, the average X-axis acceleration and the changing rate were used as features, reporting a minimum accuracy of 88.7% [53].

In Reference [54], the authors used the SVM method for the classification of the different types of vehicles with the Zero Crossing Rate (ZCR), MFCC, Spectral centroid and Spectral flux as features extracted from the microphone, reporting a minimum accuracy equal to 78.95%.

The Adaboost method was proposed in Reference [55] with the maximum, minimum, mean, standard deviation, Root Mean Square (RMS), ZCR, bandwidth, normalized phase deviation and MFCC as features collected using the microphone, gyroscope and magnetometer to identify meals, cooking, TV viewing and conversations, reporting a minimum accuracy of 65%.

The authors of Reference [56] used the J48 decision tree for the recognition of chatting, coding, writing documents, and playing games, reporting 95% accuracy with the maximum, minimum and mean as features.

In Reference [57], the cycling activity was recognized with Weka (REPTree), reporting an accuracy of 97.4% with frequency spectrum as a feature.

Other studies have been done but they used big data and distributed systems and our proposal consists of the use of local processing for the recognition of ADL and its environments [58–60].

Table 1 present the ADL and environments identified using the microphone, verifying that the standing activities are well differentiated with acoustic data.

Table 1. Activities of Daily Living (ADL) and environments identified in the literature review.

ADL:	**Number of Studies:**
Street with emergency vehicles (police, fire department and ambulance)	6
Sleeping; walking; standing; street traffic; ocean	5
Driving; river	4
Sitting; cleaning with a vacuum cleaner; train; nature; typing; dog barking; baby crying; raining; music	3
Running; lying; going upstairs; going downstairs; drinking; shopping; travelling by car; cooking; watching television; eating; working on a computer; reading; washing dishes; restaurant; laughing; door opening/closing; telephone ringing; helicopter; speech; coffee machine; elevator	2
social interaction activities; jogging; cycling; cleaning table; going to toilet; waiting in a queue; being a bar; casino; playground; clearing the throat; relaxing; coughing; sniffling; talking; grains falling; whistle; sneezing; clock ticking; arguing; football; shaver; bird; dishwasher; brushing teeth; bus; calling; air conditioner; car horn; children playing; drilling; meeting; chatting; shower; clapping; smoking alarm; hand washing	1

Based on the previous studies, the features used for the recognition of ADL and environments with acoustic data are presented in Table 2, showing that the MFCC, zero crossing rate, spectral roll-off, spectral centroid, spectral flux, total RMS energy, mean, standard deviation, minimum, median and low energy frame rate are used in more than 3 studies, with more relevance for MFCC.

At the end, the ADL and environment identification can be executed using several methods shown in Table 3. We found that the approaches with the highest accuracy are ANN, k-NN, Gradient Boosting Decision Tree, IBk lazy algorithm, logistic regression, linear regression and FNN. Following the methods for ADL and environment identification using the acoustic signal, an average accuracy higher than 90% is reported. Moreover, the method that presents better accuracy for ADL and environment the recognition is the MLP, presenting 96% accuracy (average).

Table 2. Features identified in the literature review.

Features:	Number of Studies:
Mel-Frequency Cepstral Coefficients (MFCC)	21
zero-crossing rate	8
spectral roll-off	6
spectral centroid; spectral flux	5
total Root-Mean-Square (RMS) energy	4
Mean; standard deviation; minimum; median; low energy frame rate	3
spectral spread; log power; skewness; kurtosis; sound pressure level (SPL); bandwidth; Relative Spectral Entropy (RSE)	2
total spectrum power; sub-band powers; range; angular degree; slope; coefficient of variation; inverse coefficient of variation; trimmed mean; percentiles (1st, 57th, 95th and 99th); spectral variance; spectral skewness; spectral kurtosis; spectral slope; maximum; quartiles (1st and 3rd); interquartile range; number of peaks; mean distance of peaks; mean amplitude of peaks; mean crossing rate; linear regression slope; spectral flatness; threshold; noise level; one-third-octave spectra; statistical indices; motif; normalized weighted phase deviation; Normalized Mel-Frequency Bands; short-time energy; temporal centroid; energy entropy; autocorrelation; spectral entropy	1

Table 3. Classification methods identified in the literature review.

Methods:	Number of Studies:	Average of Reported Accuracy:
Multi-Layer Perceptron (MLP)	3	96%
k-Nearest Neighbour (k-NN)	3	95%
Gradient Boosting Decision Tree	1	92%
IBk lazy algorithm	1	91%
logistic regression	1	90%
linear regression	1	90%
Feedforward Neural Networks (FNN)	1	90%
Hidden Markov Models (HMM)	2	87%
diverse density (DD)	1	87%
expected maximization (EM)	1	87%
J48 decision tree	2	84%
FT decision tree	2	84%
LMT decision tree	2	84%
Support Vector Machine (SVM)	10	77%
Gaussian mixture model (GMM)	5	76%
Deep Neural Networks (DNN)	3	68%
Linear Discriminant Classifier (LDC)	1	67%
Random Forests	3	66%
Adaboost	1	65%
Recurrent Neural Networks (RNN)	1	24%

3. Methods

In this work, we propose a model for the detection and recognition of the environment detection. This model is based on acoustic sensors and a model for the recognition of standing activities based on motion and magnetic sensors as an enhancement of a previous developed framework for the recognition of ADL and their environments [4–7,16]. The framework was designed to recognize the following ADL—running, walking, going upstairs, sleeping, going downstairs, sleeping, watching TV and standing. In addition, the following scenarios are also recognized by the framework—bar, classroom, gym, kitchen, library, street, hall, living room and bedroom.

3.1. Data Acquisition

The data acquisition module aims to capture all the sensors' data, including accelerometer, magnetometer, gyroscope and microphone. Unlike the microphone, the data from which are saved in a raw forma, this data was acquired at the same time as the study available in Reference [4] and with the same individuals.

3.2. Data Processing

On the one hand, environment recognition comprehends the use of the microphone with the application of the Fast Fourier Transform (FFT) [61] to extract the relevant features. After the application of the FFT, several features were extracted, including 26 MFCC coefficients and standard deviation, average, maximum value, minimum value, variance and median of the raw signal.

On the other hand, the recognition of the standing activities makes use of the environment recognized and accelerometer, magnetometer and/or gyroscope sensors' data with the application of a low pass filter [62], extracting the same features presented in Reference [4].

3.3. Data Fusion

This module encompasses several databases obtained from the combination of different sensors, and features, which are depicted in Figure 1. The different combinations of sensors are:

- Microphone for the Environment Detection
- Accelerometer data plus Environment Recognized
- Accelerometer and Magnetometer data plus Environment Recognized
- Accelerometer, Magnetometer and Gyroscope data plus Environment Recognized

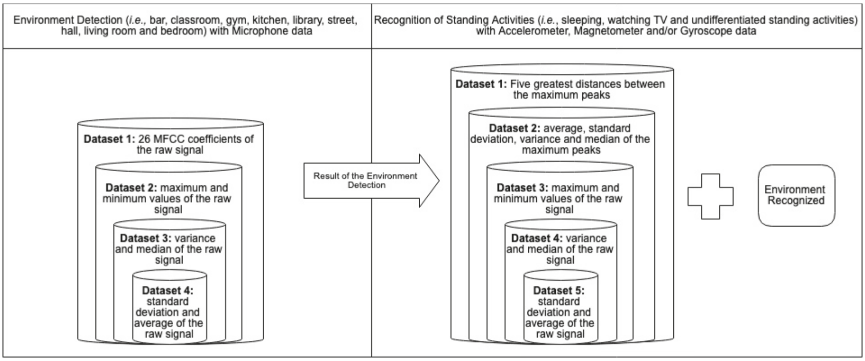

Environment Detection (i.e., bar, classroom, gym, kitchen, library, street, hall, living room and bedroom) with Microphone data
Recognition of Standing Activities (i.e., sleeping, watching TV and undifferentiated standing activities) with Accelerometer, Magnetometer and/or Gyroscope data
Dataset 1: 26 MFCC coefficients of the raw signal
Dataset 2: maximum and minimum values of the raw signal
Dataset 3: variance and median of the raw signal
Dataset 4: standard deviation and average of the raw signal
Result of the Environment Detection
Dataset 1: Five greatest distances between the maximum peaks
Dataset 2: average, standard deviation, variance and median of the maximum peaks
Dataset 3: maximum and minimum values of the raw signal
Dataset 4: variance and median of the raw signal
Dataset 5: standard deviation and average of the raw signal
Environment Recognized

Figure 1. Different combinations of features for the recognition of environment and standing activities.

3.4. Classification

This study aims to recognize nine environments, including bar, classroom, gym, kitchen, library, street, hall, living room and bedroom using the same methods and implementations, which are implemented and tested in Reference [4]. The different implementations were performed with non-normalized and normalized data, implementing a stop criterion related to the maximum number of training interactions tested with three limits, namely: 10^6, 2 × 10^6 and 4 × 10^6.

4. Results

4.1. Identification of the Environment of the Activities of Daily Living with Microphone

The implementation of MLP with Backpropagation reported the results presented in Figure 2, verifying that the accuracy reported is very low with all datasets. With non-normalized data (Figure 2a, the results achieved are between 10% and 15%. With normalized data (Figure 2b, the results obtained are between 10% and 20%, where the best results are achieved with dataset 1.

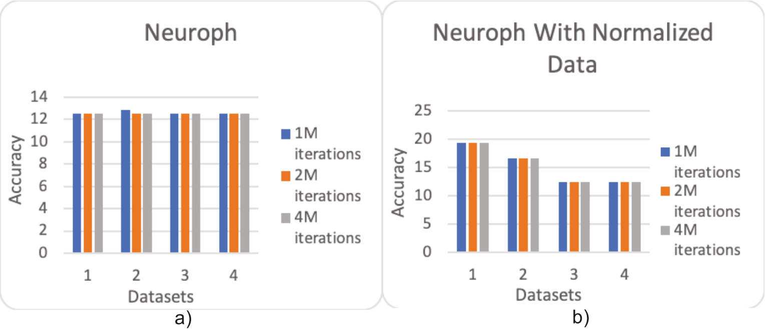


Figure 2. Results obtained with Multilayer Perceptron (MLP) with Backpropagation for the different datasets of microphone data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

Moreover, the results reported by the implementation of the FNN with Backpropagation are presented in Figure 3. In general, this implementation reports better results with non-normalized data. With non-normalized data (Figure 3a), the FNN reports results higher than 70% with dataset 1 with a maximum number of training iterations, dataset 2 with 10^6 of training iterations, and dataset 4 with 4 × 10^6 of training iterations. With normalized data (Figure 3b), the FNN reports results below than 60% but the results achieved are higher than 60% with the dataset 4 trained over 10^6 and 2 × 10^6 of iterations.

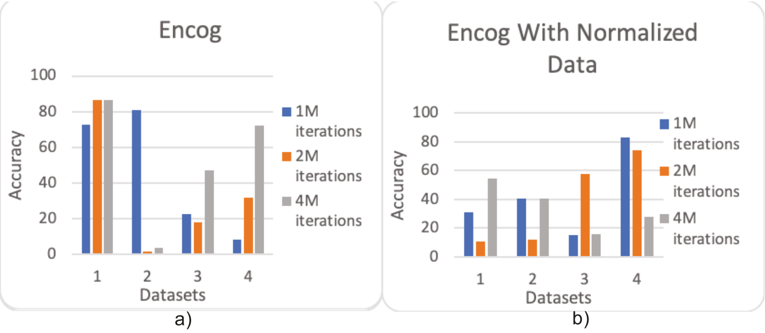


Figure 3. Results obtained with Feedforward Neural Network (FNN) with Backpropagation for the different datasets of microphone data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

The results of the implementation of DNN are presented in Figure 4, where, with non-normalized data (Figure 4a), the results obtained are below 20% with datasets 1 and 2, and the results obtained are higher than 40% with datasets 3 and 4. In addition, with normalized data (Figure 4b), the results reported are round 50% with all datasets.

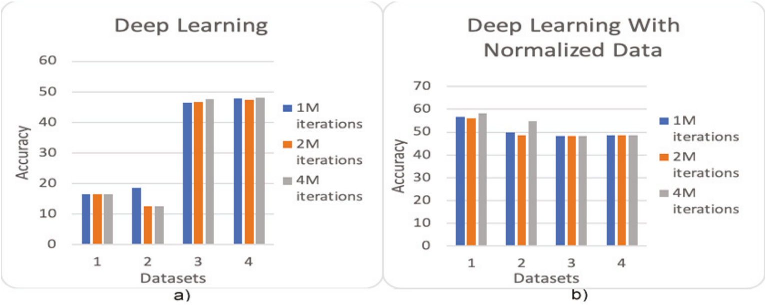


Figure 4. Results obtained with Deep Neural Network (DNN) for the different datasets of microphone data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

In Table 4, the maximum accuracies achieved with the different implementations of ANN are related to the different datasets used for the microphone data and the maximum number of training iterations, verifying that the best results are achieved with the FNN with Backpropagation with non-normalized data.

Table 4. Best accuracies obtained with the different frameworks, datasets and number of iterations for the recognition of environments using microphone data.

	Framework	**Datasets**	**Iterations Needed for Training**	**Best Accuracy Achieved (%)**
Non-normalized data	**MLP with Backpropagation**	2	10^6	12.86
	FNN with Backpropagation	1	2×10^6	86.50
	DNN	4	4×10^6	48.11
Normalized data	**MLP with Backpropagation**	1	10^6	19.43
	FNN with Backpropagation	4	10^6	82.75
	DNN	4	4×10^6	48.74

In conclusion, the method for the recognition of the environment that should be implemented in the framework for the recognition of ADL and their environments is the FNN with Backpropagation using non-normalized data, because it achieves results around 86.50% with the dataset 1.

4.2. Identification of the Standing Activities with the Environment Recognized and the Accelerometer Sensor

The use of normalized data resulted in the achievement of an accuracy of 100% with MLP with Backpropagation, FNN with Backpropagation and DNN methods, because the use of the correct recognition of environments with acoustic data provides a correct discretization of the accelerometer data.

Following the use of non-normalized data, Figure 5 shows the results obtained with MLP with Backpropagation, FNN with Backpropagation and DNN methods. MLP with Backpropagation (Figure 5a) reported results between 50% and 100%, where the better accuracy was achieved with the datasets 1 and 4. FNN with Backpropagation (Figure 5b) reported results around 100%, except with dataset 1 that achieves an accuracy around 50%. DNN method (Figure 5c) reported results around 100% with datasets 2, 4 and 5 with all training iterations, and with dataset 3 with 4 × 10^6 iterations, but the results obtained with other combinations are below expectations.

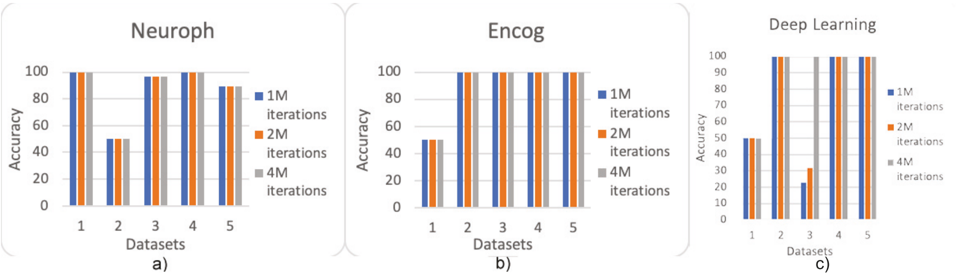


Figure 5. Results obtained with MLP with Backpropagation (**a**), FNN with Backpropagation (**b**) and DNN (**c**) methods for the different datasets of environment and accelerometer data.

In Table 5, the maximum accuracies achieved with the different types of ANN are presented with the relation of the different datasets used for the environment recognized and the accelerometer data and the maximum number of iterations.

Table 5. Best accuracies obtained with the different frameworks, datasets and number of iterations for the recognition of standing activities with the accelerometer data and the environments recognized.

	Framework	Datasets	Iterations Needed for Training	Best Accuracy Achieved (%)
Non-normalized data	**MLP with Backpropagation**	1	10^6	100.00
	FNN with Backpropagation	2	10^6	100.00
	DNN	2	10^6	100.00
Normalized data	**MLP with Backpropagation**	1	10^6	100.00
	FNN with Backpropagation	1	10^6	100.00
	DNN	1	10^6	100.00

Regarding the results obtained, in the case of the use of the environment recognized and the accelerometer data in the module for the recognition of standing activities in the framework for the identification ADL and their environments, the implementation that should be used is a DNN with normalized data because the results obtained are always 100%.

4.3. Identification of the Standing Activities with the Environment Recognized and the Accelerometer and Magnetometer Sensors

The use of normalized data resulted in the achievement of an accuracy of 100% with MLP with Backpropagation, FNN with Backpropagation and DNN methods, because the use of the correct recognition of environments with acoustic data provides a correct discretization of the accelerometer and magnetometer data.

Following the use of non-normalized data, Figure 6 shows the results obtained with MLP with Backpropagation, FNN with Backpropagation and DNN methods. MLP with Backpropagation (Figure 6a) reported results around 100%, except with the datasets 1 and 5 which achieved an accuracy around 50%. FNN with Backpropagation (Figure 6b) reported results around 100%. DNN method (Figure 6c) reported results around 100% with dataset 5 with all training iterations, and with dataset 4 with 10^6 of training iterations, but the results obtained with other combinations are below expectations.

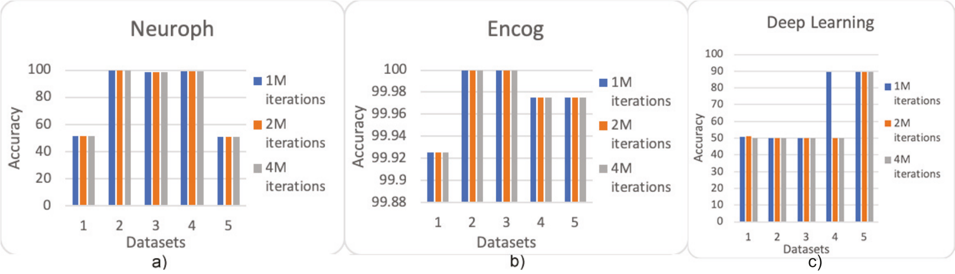


Figure 6. Results obtained with MLP with Backpropagation (**a**), FNN with Backpropagation (**b**) and DNN (**c**) methods for the different datasets of environment and accelerometer and magnetometer sensors' data.

In Table 6, the maximum accuracies achieved with the different implementations of ANN are presented with the relationship between the different datasets used for the environment recognized, and the accelerometer and magnetometer sensors' data, and the maximum number of iterations.

Table 6. Best accuracies obtained with the different frameworks, datasets and number of iterations for the recognition of standing activities with the accelerometer and magnetometer data, and the environments recognized.

	Framework	Datasets	Iterations Needed for Training	Best Accuracy Achieved (%)
Non-normalized data	**MLP with Backpropagation**	4	10^6	99.05
	FNN with Backpropagation	2	10^6	100.00
	DNN	3	10^6	89.55
Normalized data	**MLP with Backpropagation**	1	10^6	100.00
	FNN with Backpropagation	1	10^6	100.00
	DNN	1	10^6	100.00

DNN with normalized data always reported results equal to 100% with the use of the accelerometer and magnetometer sensors' data combined with the environment recognized. Thus, the framework for the identification ADL and their environments should implement the DNN with normalized data.

4.4. Identification of the Standing Activities with the Environment Recognized and the Accelerometer, Magnetometer and Gyroscope Sensors

On the one hand, the results reported by the implementation of the MLP with Backpropagation using the MLP with Backpropagation are presented in Figure 7. With non-normalized data (Figure 7a), the results achieved are around 100%, except with the datasets 1 that achieves an accuracy around 50%. With normalized data (Figure 7b), the results obtained are always around 100% with all datasets.

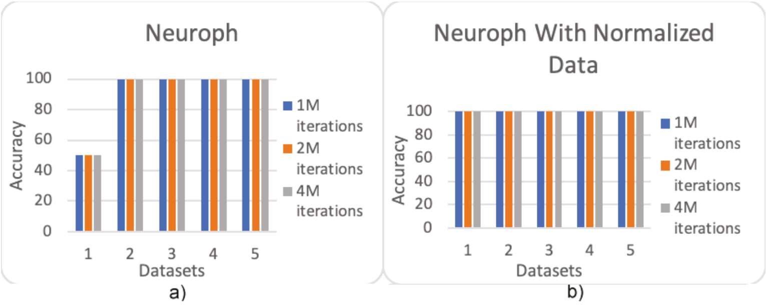


Figure 7. Results obtained with MLP with Backpropagation for the different datasets of environment, and accelerometer, magnetometer and gyroscope sensors' data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

On the other hand, the results reported by the implementation of the FNN with Backpropagation are presented in Figure 8. With non-normalized data (Figure 8a), the results achieved are always around 100%. With normalized data (Figure 8b), the results obtained are always around 100% with all datasets.

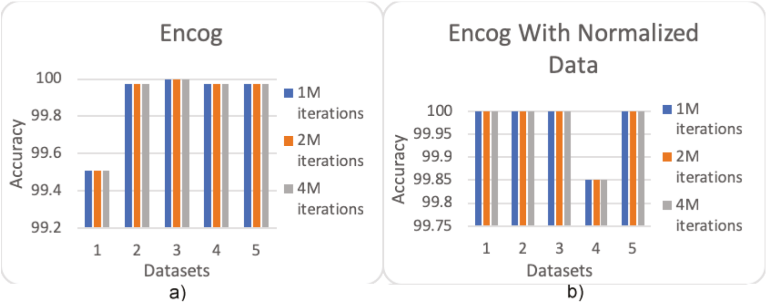


Figure 8. Results obtained with FNN with Backpropagation for the different datasets of environment and accelerometer, magnetometer and gyroscope sensors' data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

Additionally, the results reported by the implementation of DNN are presented in Figure 9. On the one hand, with non-normalized data (Figure 9a), the results obtained are around 90% with dataset 5 with all training iterations. However, the results obtained with other datasets are below the expectations. On the other hand, with normalized data (Figure 9b), the results obtained are always around 100% with all datasets.

The datasets acquired from the accelerometer, magnetometer and gyroscope combined with the environment recognized, the maximum number of iterations and the maximum accuracies reported by the different implementations of ANN are presented in Table 7.

Using the environment recognized and the accelerometer, magnetometer and gyroscope sensors' data in the module for the recognition of standing activities in the framework for the identification ADL and their environments, the reported results are always 100% with implementation of DNN with normalized data.

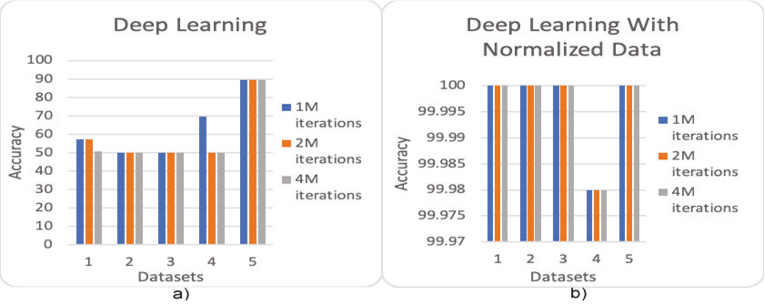


Figure 9. Results obtained with DNN for the different datasets of environment, and accelerometer, magnetometer and gyroscope sensors' data. (**a**) shows the results with non-normalized data. (**b**) shows the results with normalized data.

Table 7. Best accuracies obtained with the different frameworks, datasets and number of iterations for the recognition of standing activities with the accelerometer, gyroscope and magnetometer data, and the environments recognized.

	Framework	Datasets	Iterations Needed for Training	Best Accuracy Achieved (%)
Non-normalized data	**MLP with Backpropagation**	2	10^6	100.00
	FNN with Backpropagation	3	10^6	100.00
	DNN	5	10^6	89.55
Normalized data	**MLP with Backpropagation**	1	10^6	100.00
	FNN with Backpropagation	1	10^6	100.00
	DNN	1	10^6	100.00

5. Discussion

This research is included in the development of the framework for the recognition of ADL and their environments, presented in References [5–7]. Furthermore, this study is composed by several modules such as data acquisition, data processing, data fusion, and classification methods. The definition of the method for the identification started in the previous studies [4,16]. These studies have used accelerometer, gyroscope and magnetometer sensors to identify several activities such as going downstairs, going upstairs, running, walking and standing with the DNN, data normalization and L_2 regularization. In Section 4.1, the results of the recognition of the environments using the microphone data, where the environments recognized are bar, classroom, gym, kitchen, library, street, hall, living room and bedroom with the FNN with non-normalized data are presented. Fusing the environment recognized with the accelerometer, gyroscope and magnetometer sensors' data, the recognition of more standing activities (i.e., watching TV and sleeping) was allowed, increasing the number of ADL recognized at this stage of the development of the framework for the recognition of ADL and environments, as presented in Figure 10.

The characteristics of the mobile devices, that is, the number of sensors available, influences the methods for data fusion and artificial intelligence chosen. Ideally, all sensors available in the mobile device should be used to increase the accuracy of the method. In Figure 10, a simplified schema for the development of a framework for the identification of ADL is presented.

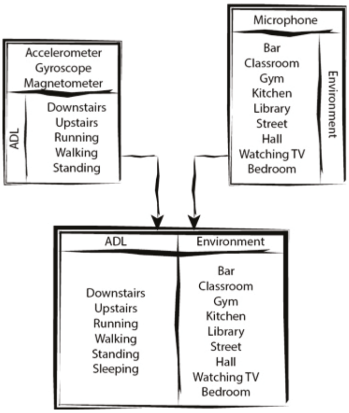


Figure 10. ADL and environments recognized by the framework for the recognition of ADL and environments.

Based on the results reported, the use of acoustic data revealed results with low accuracy because, due to the amount of data used, it reports that the ANN are overfitted. In order to avoid the overfitting problem, we used the early-stop technique, stopping the training of the ANN, when the reducing of the training error stopped. The recognition of standing activities includes only the results obtained with the recognition of the environment. The results obtained for the recognition of standing activities are around 100%, because we considered that the environment is correctly recognized. The results of the final framework will be different because of the recognition of environments that reported lower accuracy. This study only took into account the recognition of environments and standing activities separately. The use of the environment recognized correctly distinguish the activity performed.

The implementation of the framework for the recognition of ADL and their environments is composed by data acquisition, data processing, data cleaning, feature extraction, data fusion and data classification methods. Firstly, based on the results obtained in Section 4.1, the best results achieved for each implementation are presented in Table 4. The best method for the recognition of the environments is the FNN with non-normalized data, reporting an accuracy of 86.50%. Secondly, based on results obtained with the use of the environment recognized and the accelerometer data, presented in Section 4.2, the recognition of standing activities is allowed and the best results achieved for each implementation are presented in Table 4. The best method for the recognition of the standing activities is the DNN with normalization of the data and the application of L_2 regularization, reporting an accuracy of 100%. Thirdly, based on results obtained with the use of the environment recognized and the accelerometer and magnetometer sensors' data, presented in Section 4.3, the recognition of standing activities is allowed and the best results achieved for each implementation are presented in Table 5. The best method for the recognition of the standing activities is the DNN with normalization of the data and the application of L_2 regularization, reporting an accuracy of 100%. Finally, based on results obtained with the use of the environment recognized and the accelerometer, magnetometer and gyroscope sensors' data, presented in Section 4.4, the recognition of standing activities is allowed and the best results achieved for each implementation are presented in Table 6. The best method for the recognition of standing activities is the DNN with normalization of the data and the application of L_2 regularization, reporting an accuracy of 100%.

Our results and implementations cannot be directly compared with other studies because the datasets and implementation code used by other authors are not share. We asked other authors about the details of the implementation but they did not answer at the moment.

In conclusion, when the activity was recognized as standing and the environment is correctly identified, the accuracy for the recognition of standing activities is 100%. At this stage of the framework for the recognition of ADL and their environments, two different classification methods are defined, these are:

- DNN with normalized data for the general identification of ADL;
- FNN with non-normalized data for the general identification of the environments;
- DNN with normalized data for the identification of standing activities.

6. Conclusions

The development of a framework for ADL [1] and environment recognition using mobile sensors, including accelerometer, gyroscope, magnetometer and microphone, with the architecture presented in References [5–7], has several steps including data acquisition, data processing, data fusion and classification methods. At this stage of the development, the proposed identified ADL are running, walking, standing, going downstairs and upstairs, and sleeping, and the proposed identified environments are bar, classroom, gym, kitchen, library, street, hall, watching TV and bedroom.

Depending on the types of sensors, several features were extracted from the sensors' data for further processing. The features extracted from the microphone are 26 MFCC coefficients and standard deviation, average, maximum value, minimum value, variance and median of the raw signal. Following the motion and magnetic sensors, we extracted the same features of the previous study [4]. The method developed should be adapted to the number of sensors available in the off-the-shelf mobile devices and adapted to the limited resources of these devices.

In coherence with the previous studies [4,16], this research includes the comparison of three different implementations of ANN, such as MLP and FNN with Backpropagation, and the DNN. The DNN is the best method for the recognition of general ADL and standing activities, but the FNN with Backpropagation is the best method for the recognition of environments. In Reference [4], the different parameters of the ANN implemented are detailed.

The accuracies of the recognition ADL and their environments are different depending on the different stages of the framework for the recognition of ADL and environments. Firstly, the best accuracy for the recognition of the general ADL, presented in previous studies [4,16], is 85.89%, implementing the DNN using L_2 regularization and normalized data. Secondly, the best accuracy for the recognition of the environments is 86.50%, implementing the FNN with Backpropagation using non-normalized data. Finally, the recognition of standing activities are always around 100% with all implementations studied, but, due to the performance, the best method for the implementation in the framework is the DNN using L_2 regularization and normalized data.

As future work, we intend to develop a framework for the identification of ADL and their environments, adapting the method to the number of sensors available on the mobile device. The recognition of the environments allows the framework for identifying the location in the indoor/outdoor environments, where the ADL were performed. The environment recognition can also improve the recognition of ADL, increasing the number of ADL recognized. The data related to this research are available in a free repository [63].

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, writing—original draft preparation, writing—review and editing: I.M.P., G.M., N.M.G., N.P., F.F.R., S.S., M.C.T. and E.Z.

Funding: This work is funded by FCT/MEC through national funds and co-funded by FEDER-PT2020 partnership agreement under the project **UID/EEA/50008/2019**.

Acknowledgments: This work is funded by FCT/MEC through national funds and when applicable co-funded by FEDER-PT2020 partnership agreement under the project **UID/EEA/50008/2019** This work is funded by FCT/MEC through national funds and co-funded by FEDER-PT2020 partnership agreement under the project (*Este trabalho é financiado pela FCT/MEC através de fundos nacionais e cofinanciado pelo FEDER, no âmbito do Acordo de Parceria PT2020 no âmbito do projeto UID/EEA/50008/2019*). This article is based upon work from COST Action IC1303-AAPELE—Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA16226–SHELD-ON—Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information in www.cost.eu.

Conflicts of Interest: The authors declare no comflicts of interest.

References

1. Foti, D.; Koketsu, J.S. Activities of daily living. In *Pedretti's Occupational Therapy: Practical Skills for Physical Dysfunction*; Elsevier: Amsterdam, The Netherlands, 2013; Volume 7, pp. 157–232.
2. Salazar, L.H.A.; Lacerda, T.; Nunes, J.V.; von Wangenheim, C.G. A Systematic Literature Review on Usability Heuristics for Mobile Phones. *Int. J. Mob. Hum. Comput. Interact.* **2013**, *5*, 50–61. [CrossRef]
3. Garcia, N.M. *A Roadmap to the Design of A Personal Digital Life Coach*; Springer: Berlin, Germany, 2016.
4. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C. Identification of Activities of Daily Living through Data Fusion on Motion and Magnetic Sensors embedded on Mobile Devices. *Pervasive Mob. Comput.* **2018**, *47*, 78–93. [CrossRef]
5. Pires, I.; Garcia, N.; Pombo, N.; Flórez-Revuelta, F. From Data Acquisition to Data Fusion: A Comprehensive Review and a Roadmap for the Identification of Activities of Daily Living Using Mobile Devices. *Sensors* **2016**, *16*, 184. [CrossRef] [PubMed]
6. Pires, I.M.; Garcia, N.M.; Flórez-Revuelta, F. Multi-sensor data fusion techniques for the identification of activities of daily living using mobile devices. In Proceedings of the ECMLPKDD 2015 Doctoral Consortium, European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Porto, Portugal, 7–11 September 2015.
7. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Identification of Activities of Daily Living Using Sensors Available in off-the-shelf Mobile Devices: Research and Hypothesis. In Proceedings of the Ambient Intelligence-Software and Applications-7th International Symposium on Ambient Intelligence (ISAmI 2016), Seville, Spain, 1–3 June 2016; pp. 121–130.
8. Banos, O.; Damas, M.; Pomares, H.; Rojas, I. On the use of sensor fusion to reduce the impact of rotational and additive noise in human activity recognition. *Sensors* **2012**, *12*, 8039–8054. [CrossRef] [PubMed]
9. Akhoundi, M.A.A.; Valavi, E. Multi-Sensor Fuzzy Data Fusion Using Sensors with Different Characteristics. *arXiv* **2010**, arXiv:1010.6096.
10. Paul, P.; George, T. An Effective Approach for Human Activity Recognition on Smartphone. In Proceedings of the 2015 IEEE International Conference on Engineering and Technology (Icetech), Coimbatore, India, 20 March 2015; pp. 45–47. [CrossRef]
11. Hsu, Y.-W.; Chen, K.-H.; Yang, J.-J.; Jaw, F.-S. Smartphone-based fall detection algorithm using feature extraction. In Proceedings of the 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15–17 October 2016; pp. 1535–1540.
12. Dernbach, S.; Das, B.; Krishnan, N.C.; Thomas, B.L.; Cook, D.J. Simple and Complex Activity Recognition through Smart Phones. In Proceedings of the 8th International Conference on Intelligent Environments (IE), Guanajuato, Mexico, 26–29 June 2012; pp. 214–221.
13. Shen, C.; Chen, Y.F.; Yang, G.S. On Motion-Sensor Behavior Analysis for Human-Activity Recognition via Smartphones. In Proceedings of the IEEE International Conference on Identity, Security and Behavior Analysis (Isba), Sendai, Japan, 29 February–2 March 2016; pp. 1–6.
14. Wang, D. Pattern recognition: Neural networks in perspective. *IEEE Expert* **1993**, *8*, 52–60. [CrossRef]
15. Doya, K.; Wang, D. Exciting Time for Neural Networks. *Neural Netw.* **2015**, *61*. [CrossRef]
16. Pires, I.M.; Garcia, N.M.; Pombo, N.; Pires, F.FL.; Spinsante, S.; Teixeira, M.C.; Zdravevski, E. Pattern Recognition Techniques for the Identification of Activities of Daily Living using Mobile Device Accelerometer. *PeerJ Prepr.* **2019**. [CrossRef]
17. Gripenberg, G. Approximation by neural networks with a bounded number of nodes at each level. *J. Approx. Theory* **2003**, *122*, 260–266. [CrossRef]
18. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef]
19. Costarelli, D.; Vinti, G. Pointwise and uniform approximation by multivariate neural network operators of the max-product type. *Neural Netw.* **2016**, *81*, 81–90. [CrossRef] [PubMed]
20. Lane, N.D.; Mohammod, M.; Lin, M.; Yang, X.; Lu, H.; Ali, S.; Doryab, A.; Berke, E.; Choudhury, T.; Campbell, A. Bewell: A smartphone application to monitor, model and promote wellbeing. In Proceedings of the 5th international ICST conference on pervasive computing technologies for healthcare, Dublin, Ireland, 23–26 May 2011.

21. Mengistu, Y.; Pham, M.; Do, H.M.; Sheng, W. AutoHydrate: A Wearable Hydration Monitoring System. In Proceedings of the IEEE/Rsj International Conference on Intelligent Robots and Systems (Iros 2016), Daejeon, Korea, 9–14 October 2016; pp. 1857–1862. [CrossRef]
22. Nishida, M.; Kitaoka, N.; Takeda, K. Daily activity recognition based on acoustic signals and acceleration signals estimated with Gaussian process. In Proceedings of the 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 16–19 December 2015; pp. 279–282.
23. Filios, G.; Nikoletseas, S.; Pavlopoulou, C.; Rapti, M.; Ziegler, S. Hierarchical Algorithm for Daily Activity Recognition via Smartphone Sensors. In Proceedings of the IEEE 2nd World Forum on Internet of Things (Wf-Iot), Milan, Italy, 14–16 December 2015; pp. 381–386. [CrossRef]
24. Delgado-Contreras, J.R.; Garæia-Vázquez, J.P.; Brena, R.F.; Galván-Tejada, C.E.; Galván-Tejada, J.I. Feature Selection for Place Classification through Environmental Sounds. *Procedia Comput. Sci.* **2014**, *37*, 40–47. [CrossRef]
25. Rahman, T.; Adams, A.T.; Zhang, M.; Cherry, E.; Zhou, B.; Peng, H.; Choudhury, T. BodyBeat: A mobile system for sensing non-speech body sounds. In Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, Bretton Woods, NH, USA, 16–19 June 2014.
26. Mielke, M.; Brück, R. Smartphone application for automatic classification of environmental sound. In Proceedings of the 20th International Conference Mixed Design of Integrated Circuits and Systems-MIXDES, Gdynia, Poland, 20–22 June 2013; pp. 512–515.
27. Guo, X.; Toyoda, Y.; Li, H.; Huang, J.; Ding, S.; Liu, Y. Environmental sound recognition using time-frequency intersection patterns. In Proceedings of the 3rd International Conference on Awareness Science and Technology (iCAST), Ypsilanti, MI, USA, 3–5 October 2011; pp. 243–246.
28. Pillos, A.; Alghamidi, K.; Alzamel, N.; Pavlov, V.; Machanavajhala, S. A real-time environmental sound recognition system for the Android OS. In Proceedings of the Detection and Classification of Acoustic Scenes and Events, Budapest, Hungary, 3 September 2016.
29. Mielke, M.; Brueck, R. Design and evaluation of a smartphone application for non-speech sound awareness for people with hearing loss. In Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milan, Italy, 25–29 August 2015; pp. 5008–5011.
30. Dubey, H.; Mehl, M.R.; Mankodiya, K. BigEAR: Inferring the Ambient and Emotional Correlates from Smartphone-Based Acoustic Big Data. In Proceedings of the IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), Washington, DC, USA, 27–29 June 2016; pp. 78–83.
31. Lane, N.D.; Georgiev, P.; Qendro, L. DeepEar: Robust smartphone audio sensing in unconstrained acoustic environments using DNN. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Osaka, Japan, 7–11 September 2015.
32. Wang, J.; Ruby, R.; Wang, L.; Wu, K. Accurate Combined Keystrokes Detection Using Acoustic Signals. In Proceedings of the 12th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN), Shenyang, China, 9–14 December 2016.
33. Rossi, M.; Feese, S.; Amft, O.; Braune, N.; Martis, S.; Tröster, G. AmbientSense: A real-time ambient sound recognition system for smartphones. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), San Diego, CA, USA, 18–22 March 2013; pp. 230–235.
34. Nishijima, K.; Uenohara, S.; Furuya, K. A Study on the Optimum Number of Training Data in Snore Activity Detection Using SVM. In Proceedings of the 10th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS), Fukuoka, Japan, 6–8 July 2016; pp. 582–584.
35. Nishijima, K.; Uenohara, S.; Furuya, K. Snore activity detection using smartphone sensors. In Proceedings of the IEEE International Conference on Consumer Electronics-Taiwan, Taipei, Taiwan, 6–8 June 2015; pp. 128–129.
36. Gaunard, P.; Mubikangiey, C.G.; Couvreur, C.; Fontaine, V. Automatic classification of environmental noise events by hidden Markov models. *IEEE Int. Conf. Acoust. Speech Signal Process.* **1998**, *3*, 3609–3612.
37. Zilli, D.; Parson, O.; Merrett, G.V.; Rogers, A. A Hidden Markov Model-Based Acoust. Cicada Detect. Crowdsourced Smartphone Biodivers. Monit. *J. Artif. Int. Res.* **2014**, *51*, 805–827.

38. Song, T.; Cheng, X.; Li, H.; Yu, J.; Wang, S.; Bie, R. Detecting driver phone calls in a moving vehicle based on voice features. In Proceedings of the IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016.
39. Chen, Y.A.; Chen, J.; Tseng, Y.C. Inference of Conversation Partners by Cooperative Acoustic Sensing in Smartphone Networks. *IEEE Trans. Mob. Comput.* **2016**, *15*, 1387–1400. [CrossRef]
40. Gomes, E.F.; Batista, B.; Jorge, P.M. Using Smartphones to Classify Urban Sounds. In Proceedings of the Ninth International Conference on Computer Science & Software Engineering, Porto, Portugal, 20–22 July 2016.
41. Lu, H.; Pan, W.; Lane, N.D.; Choudhury, T.; Campbell, A.T. SoundSense: Scalable sound sensing for people-centric applications on mobile phones. In Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services, Kraków, Poland, 22–25 June 2009.
42. Sigtia, S.; Stark, A.M.; Krstulovic, S.; Plumbley, M.D. Automatic Environmental Sound Recognition: Performance Versus Computational Cost. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2016**, *24*, 2096–2107. [CrossRef]
43. Kelly, D.; Caulfield, B. Pervasive Sound Sensing: A Weakly Supervised Training Approach. *IEEE Trans. Cybern.* **2016**, *46*, 123–135. [CrossRef]
44. Abreha, G.T. An Environmental Audio-Based Contextrecognition System Using Smartphones. Master's Thesis, University of Twente, Enschede, The Netherlands, August 2014.
45. Saki, F.; Sehgal, A.; Panahi, I.; Kehtarnavaz, N. Smartphone-based real-time classification of noise signals using subband features and random forest classifier. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 2204–2208.
46. Inoue, S.; Ueda, N.; Nohara, Y.; Nakashima, N. Mobile activity recognition for a whole day: Recognizing real nursing activities with big dataset. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Osaka, Japan, 7–11 September 2015.
47. Bountourakis, V.; Vrysis, L.; Papanikolaou, G. Machine Learning Algorithms for Environmental Sound Recognition: Towards Soundscape Semantics. In Proceedings of the Audio Mostly 2015 on Interaction with Sound, Thessaloniki, Greece, 7–9 October 2015.
48. Cheffena, M. Fall Detection Using Smartphone Audio Features. *IEEE J. Biomed. Health Inf.* **2016**, *20*, 1073–1080. [CrossRef]
49. Sehgal, A.; Saki, F.; Kehtarnavaz, N. Real-time implementation of voice activity detector on ARM embedded processor of smartphones. In Proceedings of the 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE), Edinburgh, UK, 19–21 June 2017. [CrossRef]
50. Elhamshary, M.; Youssef, M.; Uchiyama, A.; Yamaguchi, H.; Higashino, T. CrowdMeter: Congestion Level Estimation in Railway Stations Using Smartphones. In Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom), Athens, Greece, 19–23 March 2018; pp. 1–12. [CrossRef]
51. Hoyos-Barceló, C.; Monge-Álvarez, J.; Shakir, M.Z.; Alcaraz-Calero, J.-M.; Casaseca-de-la-Higuera, P. Efficient k-NN Implementation for Real-Time Detection of Cough Events in Smartphones. *IEEE J. Biomed. Health Inform.* **2018**, *22*, 1662–1671. [CrossRef]
52. Monge-Alvarez, J.; Hoyos-Barcelo, C.; Lesso, P.; Casaseca-de-la-Higuera, P. Robust Detection of Audio-Cough Events using local Hu moments. *IEEE J. Biomed. Health Inform.* 2018, 23, 184–196. [CrossRef]
53. Bi, C.; Xing, G.; Hao, T.; Huh, J.; Peng, W.; Ma, M. FamilyLog: A mobile system for monitoring family mealtime activities. In Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications (PerCom), Seattle, WA, USA, 21–25 March 2017; pp. 21–30. [CrossRef]
54. Soni, S.; Aggarwal, N.; Vij, D.; Doegar, A. Acoustic Scene Classification for Personal Commuting Mode: Detecting Polluting vs. Non Polluting Vehicles. In Proceedings of the 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 11–12 January 2018; pp. 274–279. [CrossRef]
55. Gu, F.; Niu, J.; He, Z.; Jin, X.; Rodrigues, J.J.P.C. SmartBuddy: An Integrated Mobile Sensing and Detecting System for Family Activities. In Proceedings of the 2017 IEEE Global Communications Conference (GLOBECOM 2017), Singapore, 4–8 December 2017; pp. 1–7. [CrossRef]
56. Yu, Z.; Du, H.; Xiao, D.; Wang, Z.; Han, Q.; Guo, B. SmartBuddy: An Integrated Mobile Sensing and Detecting System for Family Activities. *IEEE Internet Things J.* **2018**, *5*, 1156–1168. [CrossRef]

57. Kawanaka, S.; Kashimoto, Y.; Firouzian, A.; Arakawa, Y.; Pulli, P.; Yasumoto, K. Approaching vehicle detection method with acoustic analysis using smartphone for elderly bicycle driver. In Proceedings of the 2017 Tenth International Conference on Mobile Computing and Ubiquitous Network (ICMU), Toyama, Japan, 3–5 October 2017; pp. 1–6. [CrossRef]
58. Su, X.; Sperlì, G.; Moscato, V.; Picariello, A.; Esposito, C.; Choi, C. An Edge Intelligence Empowered Recommender System Enabling Cultural Heritage Applications. *IEEE Trans. Ind. Inform.* **2019**, *15*, 4266–4275. [CrossRef]
59. Chen, L.; Nugent, C.D. Sensor-Based Activity Recognition Review. In *Human Activity Recognition and Behaviour Analysis*; Springer: Cham, Switzerland, 2019; pp. 23–47.
60. Amato, F.; Moscato, V.; Picariello, A.; Sperli'ì, G. Extreme events management using multimedia social networks. *Future Gener. Comput. Syst.* **2019**, *94*, 444–452. [CrossRef]
61. Rader, C.; Brenner, N. A new principle for fast Fourier transformation. *IEEE Trans. Acoust. Speech Signal Process.* **1976**, *24*, 264–266. [CrossRef]
62. Graizer, V. Effect of low-pass filtering and re-sampling on spectral and peak ground acceleration in strong-motion records. In Proceedings of the 15th World Conference of Earthquake Engineering, Lisbon, Portugal, 24–28 September 2012; pp. 24–28.
63. ALLab. August 2017-Multi-Sensor Data Fusion in Mobile Devices for the Identification of Activities of Daily Living-ALLab Signals. Available online: https://allab.di.ubi.pt/mediawiki/index.php/August_2017-_Multi-sensor_data_fusion_in_mobile_devices_for_the_identification_of_activities_of_daily_living (accessed on 2 September 2017).

electronics

Article

Robot Motion Control via an EEG-Based Brain–Computer Interface by Using Neural Networks and Alpha Brainwaves

Nikolaos Korovesis [1], Dionisis Kandris [1,*], Grigorios Koulouras [2] and Alex Alexandridis [2]

[1] microSENSES Research Laboratory, Department of Electrical and Electronic Engineering, Faculty of Engineering, University of West Attica, 12244 Athens, Greece; ee06416@uniwa.gr

[2] TelSiP Research Laboratory, Department of Electrical and Electronic Engineering, Faculty of Engineering, University of West Attica, 12244 Athens, Greece; gregkoul@uniwa.gr (G.K.); alexx@uniwa.gr (A.A.)

* Correspondence: dkandris@uniwa.gr; Tel.: +30-210-538-1545

Received: 2 November 2019; Accepted: 19 November 2019; Published: 21 November 2019

Abstract: Modern achievements accomplished in both cognitive neuroscience and human–machine interaction technologies have enhanced the ability to control devices with the human brain by using Brain–Computer Interface systems. Particularly, the development of brain-controlled mobile robots is very important because systems of this kind can assist people, suffering from devastating neuromuscular disorders, move and thus improve their quality of life. The research work presented in this paper, concerns the development of a system which performs motion control in a mobile robot in accordance to the eyes' blinking of a human operator via a synchronous and endogenous Electroencephalography-based Brain–Computer Interface, which uses alpha brain waveforms. The received signals are filtered in order to extract suitable features. These features are fed as inputs to a neural network, which is properly trained in order to properly guide the robotic vehicle. Experimental tests executed on 12 healthy subjects of various gender and age, proved that the system developed is able to perform movements of the robotic vehicle, under control, in forward, left, backward, and right direction according to the alpha brainwaves of its operator, with an overall accuracy equal to 92.1%.

Keywords: brain–computer interface (BCI); human–robot interaction; assistive robotics; motion control; electroencephalography (EEG); alpha brainwaves; neural network (NN)

1. Introduction

Communication within the body of mammals takes place via both electrical and chemical signals. *Electrophysiology* is the branch of physiology that studies the electrical activities which are associated with bodily parts. The recording of electrophysiological data is performed by placing electrodes at the corresponding areas of interest. By this method, there are numerous systems developed which are able to monitor the electrical activity and corresponding electrophysiological data in various organs such as heart, brain, eyes, muscles, and stomach [1–3].

Electroencephalography (EEG) is an electrophysiological method which is used in order to monitor the electrical activity of the brain by placing electrodes on the external surface of the scalp. EEG records variations of voltage caused by the flow of ionic current in the interior of the brain's neurons. Therefore, EEG signals are waveforms, also known as brainwaves or brain waveforms, which signify the neural oscillations produced by neurons which intercommunicate. Brainwaves are detected in the frequency domain, having signal intensity measured in microvolts (μV) and signal frequency usually ranging from 1 to 100 Hz. According to their frequency, there are specific bands classified as delta (δ) (1–4 Hz), theta (θ) (4–7 Hz), alpha (α) (8–13 Hz), beta (β) (13–30 Hz), and gamma (γ) (>30 Hz) [4].

A *Brain–Computer Interface* (BCI) is a system that enables communication between brain and machines. A BCI, in order to perform its purposes, records brain signals, interprets them, and produces corresponding commands to a connected machine [5]. BCI technology is used in various applications, such as security and authentication, education, neuromarketing and advertisement, games and entertainment, and several medical applications, such as cognitive neuroscience, brain-related prevention and diagnosis of health problems, rehabilitation, and restoration [6–9].

This article presents the development of a BCI-based system that performs the motion control of a robotic vehicle by using brainwaves of a human operator. After capturing the brainwaves via EEG, a set of features is extracted and given as input to a neural network, which is trained to predict the desired movement of the robotic vehicle. The rest of this paper is organized as follows: In Section 2, the theoretical background of the research carried out is set up. In Section 3, the structure and operation of the proposed system are explained. In Section 4, the performance of the system is evaluated through the description of the experimental tests made, and the presentation of the corresponding results and discussion on them. Finally, Section 5 concludes the article and proposes future research work.

2. Theoretical Background

2.1. BCI Types

A BCI provides an interconnection platform that supports the full duplex communication between the brain and an external device. According to the way that BCIs use to set up the brain–device interconnection, they are classified as non-invasive or invasive. *Non-invasive* BCIs use electrodes placed on the scalp. They are easy and safe to use, low-cost, portable, and offer a relatively high temporal resolution. *Invasive* BCIs use electrodes implanted in the interior of the scalp. Comparatively to non-invasive BCIs offer higher values of amplitude, spatial resolution, and resistance to noise. However, they require neurosurgery operations and they are both unsafe and expensive. Furthermore, scar tissues decrease the quality of signals received. Practically, non-invasive BCIs are used more often.

There are various non-invasive methodologies used in BCI technology, such as *Positron Emission Tomography* (PET), *functional Magnetic Resonance Imaging* (fMRI), and *Near-Infrared Spectroscopy* (NIRS), which study changes made in the blood flow, *magnetoencephalography* (MEG), which monitors the magnetic action of the brain, and EEG, which records the electric activity of the brain. Both NIRS and fMRI BCIs offer high spatial resolution, but poor temporal resolution. Moreover, MEG and PET BCIs offer high spatial and temporal resolution. However, PET BCIs require the inoculation of a radioactive constituent into the bloodstream. Furthermore, both fMRI and MEG methods rely on the use of equipment which is not only costly, but also huge. EEG BCIs are by far the most popular type, because, despite their relatively poor spatial resolution, they have high temporal resolution, low-cost, and easy installation. [6].

Moreover, BCIs are classified as either exogenous or endogenous, according to the nature of the input signals. *Exogenous* BCIs analyze the brain activity created due to external stimuli. They are easy to set up and offer high bit rates, but they need the continuous response of the user to outward incitements which may be either tiring, or even unfeasible. *Endogenous* BCIs use self-regulation of brainwaves without external stimuli. They provide lower data transfer rates but they can be operated via free self-control even by users with sensory organs affected or suffering from motor neuron diseases [10].

Similarly, BCI systems are classified, according to the method used for input data processing, as synchronous or asynchronous. *Synchronous* BCIs analyze the brain signals only after a specific prompt and during predefined time intervals. Thus, the overall process is better organized and the user is free to make any kind of movements, which would produce artifacts, when brain signals are not observed. They also require minimal training and have stable performance and high accuracy. *Asynchronous* BCIs inspect brain signals successively, thus letting the user act at free will. Therefore, they offer more natural human–machine interaction. However, they are more complex in design and evaluation and require extensive training. Moreover, their performance may vary between users, and their accuracy is not very high [10].

2.2. Brainwaves for EEG-BCIs

The most commonly used types of brain waveforms to develop EEG-based BCIs are P300, SSVEP, ErrP, ERD/ERS, and alpha brainwaves [11].

P300 is an event-related positive potential deflection which is caused by the reaction to a desired external stimulus of visual, auditory, or tactile modality. P300 waveforms are typically measured, with a latency of roughly 250 to 500 ms between stimulus and response, by using electrodes located over the parietal lobe of the scalp.

Steady state visually evoked potentials (SSVEP) are brain waveforms of exogenous type that are generated as responses to visual stimulation at specific frequencies ranging from 3.5 Hz to 75 Hz. Considering that SSVEP signals often have their highest values at medial occipital electrode sites, they are supposed to originate mostly from the primary visual cortex.

Event-related desynchronization and event-related synchronization (ERD/ERS) waves are endogenous brain signals, which are generated when performing mental tasks, such as motor imagery or mental arithmetic. They can be measured at different cortical locations.

Error-related potential (ErrP) waveforms are brain signals which are activated every time that a subject identifies the commitment of an error which has been made either by himself/herself or by another individual during various choice tasks. Waves of this kind can be captured by applying electrodes on various brain regions including the anterior cingulate cortex, anterior insula, inferior parietal lobe, and intraparietal sulcus, as well as other regions of the cortex, subcortex, and cerebellum.

Alpha brainwaves are brain signals which have their amplitude increased whenever the eyes of an individual are closed during wakeful relaxation. In contrast, the amplitude of alpha waveforms is diminished for the duration of sleepiness and sleep and also when having eyes opened while mental effort is performed. This phenomenon is usually referred to as *alpha rhythm blocking*. Alpha brain waveforms can be monitored by applying a number of electrodes on both sides of the posterior segments of the scalp where the occipital lobe, which is the center of visual processing activities in the brain, is positioned.

2.3. BCI Operation

The operation of a typical BCI system is based on the sequential execution of a number of procedures, which namely are signal acquisition, preprocessing, feature extraction, classification, translation, and feedback to operator [10,11], as shown in Figure 1.

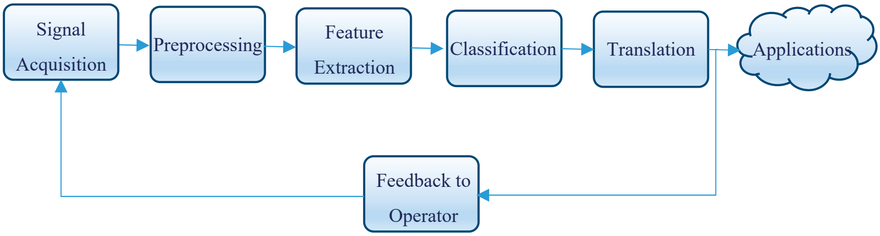


Figure 1. Block diagram representing the processes performed in a typical Brain–Computer Interface.

In EEG-BCIs, *signal acquisition* is performed by using electrodes which are positioned along the scalp of the user. Normally, the settlement of electrodes on the scalp is performed in compliance to the International 10–20 system. According to this system, electrodes are located on the scalp at 10% and 20% of a measured distance from reference spots including nasion, inion, left, and right preauricular [10].

The pattern of this system is depicted in Figure 2, where odd numbers refer to the left side of the head, even numbers refer to the right side, A1 and A2 refer to the earlobes and 'Fp', 'F', 'T', 'C', 'P', and 'O' stand for the prefrontal, frontal, temporal, central, parietal, and occipital areas of the brain, correspondingly.

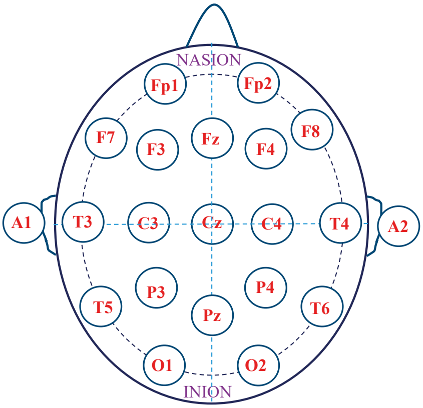


Figure 2. Top view of the international 10–20 electrode placement system on a human scalp.

Preprocessing is the procedure which is carried out in order to reduce the noise from the signal and apply some filtering and other methods in order to remove artifacts which are caused by endogenous sources, such as motions of eyes, muscles, and heart, and exogenous sources, such as power-line coupling and impedance mismatch [12]. Preprocessing is usually performed by using low-pass, high-pass, band-pass, or notch filtering. However, the use of such filters may eliminate useful elements of EEG signals having the same frequency band as artifacts [13].

In *feature extraction*, specific features of the signals in time domain or/and frequency domain that can expressively differentiate specific classes are extracted and positioned into a feature vector in order to enable the classification phase which follows. Autoregressive (AR), Hjorth, and EEG signal power are commonly used feature extraction techniques [14].

During the *classification* phase, a properly built algorithm is used. This algorithm distinguishes between classes which correspond to various brain activity patterns by deciding to which of these classes every feature vector suits best. Neural networks (NNs) are widely used as classifiers in BCIs because they provide the ability to approximate nonlinear decision boundaries [15,16]. Alternatively, linear discriminant analysis (LDA), support vector machines (SVM), and statistical classifiers may be used [17]. The advantage of LDA is that it is a simple-to-use probabilistic approach based on Bayes' Rule. On the other hand, NNs have the advantage of being able to approximate nonlinear decision boundaries. In cases where a small amount of training data is available, the use of SVM is a very good choice. Finally, statistical classifiers have the ability to represent the uncertainty that is inherent in brain signals.

During the *translation* phase the extracted signal features are converted into particular commands to the device(s) under control, through the use of dedicated translation algorithms. Specifically, these algorithms have the ability not only to adapt to the continuing variations of the signal features, but also to ensure that the complete device control range is covered by the specific signal features from the user.

Finally, in the *feedback to operator* phase, the final outcome of the overall operation of the BCI system is transferred back to the system operator, so that the performance of the system can be evaluated.

2.4. BCI-Based Robot Control

An EEG-based brain-controlled robot is a robot that uses an EEG-based BCI to receive control commands from its human operator. EEG-based brain-controlled mobile robots can support the movement of both elderly people and people who are severely disabled with destructive neuromuscular disorders, such as amyotrophic lateral sclerosis (ALS), multiple sclerosis (MS), or strokes.

There are two main classes of EEG-based brain-controlled assistive robots which namely are *brain-controlled manipulators* and *brain-controlled mobile robots*. Similarly, assistive mobile robots are classified in two categories according to their mode of operation [11].

The first category consists of assistive mobile robots which operate under *direct BCI control*. Robots of this kind are controlled exclusively via the commands that their users send to the robots controlled via BCI modules, without any additional assistance by robot intelligence elements. For this reason, they are less expensive and complex to develop and their users keep the absolute motion control.

On the other hand, the overall performance of these brain-controlled mobile robots mainly depends on the performance of the BCIs, which in many cases may have inadequate speed of response and accuracy. Furthermore, the demand for continuous production of motor control commands by the users may be extremely tiring for them.

The initial example of a robot of this kind was presented in [18] where the left and right turning movements of a robotic wheelchair were directly controlled by corresponding motion commands translated from user brain signals.

Similarly, in [19] a brain-controlled mobile robot was able to perform forward, left, and right motions by using a BCI based on motor imagery.

Moreover, in [20] the motion control of a wheelchair is performed via a BCI, which captures alpha brainwaves. Specifically, a set of icons corresponding to predefined commands are sequentially displayed on a screen and the user is able to select the desired command by closing his/her eyes as soon as its corresponding icon appears on the display unit.

The second category consists of assistive mobile robots which operate under *shared control*. In the robots of this category the control is performed by combining a BCI system along with an intelligent controller, such as an autonomous navigation system. Due to their enhanced intelligence, robots of this type are safer and less tiring for their users and more accurate in interpreting and executing their commands. On the other hand, their development is of higher cost and computational complexity.

A typical example of shared control in assistive mobile robots is proposed in [21]. In this system the operator, by using a SSVEP BCI system, has the ability to send commands in order to move a robotic wheelchair in four directions (forwards, backwards, left, and right), while an autonomous navigation system executes the delivered commands.

Similarly, in [22], by using a P300 BCI, the operator uses a list of predefined locations in order to select the desired location and then sends this selection to an autonomous navigation system, which guides a robotic wheelchair to the selected location. The limitation of the specific system is that it is able to be operated only in a known environment.

Likewise, in [23] shared control is used. Specifically, the combined use of a P300 BCI along with an autonomous navigation system is proposed in order to perform the motion control of a robotic wheelchair in an environment which is unknown. Moreover, the user has the ability to make the wheelchair turn either left or right by focusing correspondingly on one of two relative icons at a predefined visual display.

In [24] three mental tasks, which namely are the imagination of right or left hand movements and the generation of words beginning with the same random letter, were used in a BCI system applied to a robotic wheelchair. The system developed, which interacts with the user by using a PDA screen and speakers, is able to guide the robotic wheelchair both in known and unknown environments.

3. Materials and Methods

The research work carried out made use of the experimental equipment described in Section 3.1 and followed the procedure explained in Section 3.2.

3.1. Experimental Equipment

3.1.1. BCI Unit

The BCI device that was used, in order to capture the alpha brainwaves during the developed experimental procedure, is the OpenBCI Ganglion [25], which is shown in Figure 3. This board has 4 available input channels and samples data at 200 Hz.

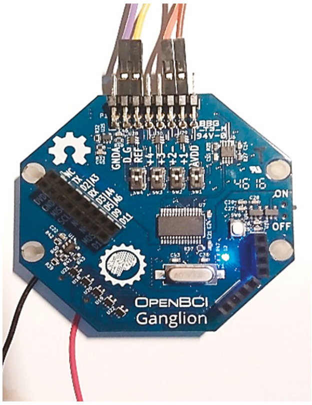


Figure 3. Overview of the Open Brain–Computer Interface (BCI) Ganglion unit used.

3.1.2. Robotic Unit

The vehicle used for the execution of the experimental procedure is a crawler robot built on Dagu Rover 5 Chassis. A Raspberry Pi (model 3 B+) acts as the central processing unit for the robot. Communication between the robotic vehicle and the computer is achieved via a TCP/IP socket connection. As soon as the classifier determines the desired movement, a command is transmitted to the robot. A serial communication is established between the Raspberry Pi and an Arduino UNO microcontroller. Once a specified command is received by the Raspberry Pi, it is relayed to the microcontroller, which in turn uses a L298N H-Bridge driver module to control the motors of the robot. The experimental platform developed is illustrated in Figure 4.

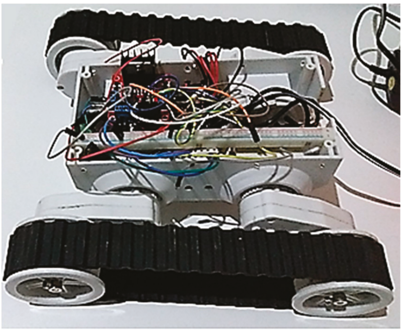


Figure 4. Overview of the robotic unit used.

3.2. Experimental Procedure

The performance of the system developed was experimentally evaluated through a series of tests. The main phases of the executed experimental procedure are as follows:

3.2.1. Signal Acquisition

The brain signals monitored are alpha waves, which, as mentioned above, is the prominent EEG wave pattern in awake adults while having eyes closed in the frequency range of 8–13 Hz. Generally, EEG-BCIs based on rhythms like alpha waveforms are less sensitive to artifacts than other types due to the fact that signal monitoring is limited in thin frequency bands. For this reason, high signal-to-noise ratio (SNR) is achieved [12].

Gold-plated electrodes were placed on the scalp of each one of the subjects that participated in the experimental procedure, according to the 10–20 system (displayed in Figure 2) at positions O1 and O2. The specific positions were chosen because, although alpha rhythms can be also generated in other parts of the brain, they are considered to exhibit greater amplitude in the posterior part of the brain, specifically at derivations O1 and O2 [26]. The reference electrode was placed on the left earlobe (A1), while the ground electrode was placed on the right earlobe (A2). In this way it is feasible to monitor alpha brainwaves.

As it was abovementioned, the amplitude of alpha brainwaves diminishes when subjects open their eyes. This is called *alpha blocking phenomenon*. By taking advantage of this phenomenon, subjects can form n-bit binary sequences by opening or closing their eyes in 2-second intervals. Each bit interval is designated by an acoustic cue.

Moreover, since this is a synchronous BCI, a button has to be pressed for the recording procedure to start. Increased alpha activity (eyes closed) corresponds to a binary ‘1’, while decreased activity (eyes open) corresponds to a binary ‘0’. As a proof of concept, 4-bit binary sequences were selected to demonstrate the effectiveness of this system. In total, 4 control signals were designated for 4 robotic movements as it can be seen in Table 1.

Table 1. Binary sequences with corresponding robotic movements.

Binary Sequence	Robotic Movement
‘1010’	Forward
‘0101’	Reverse
‘1100’	Left
‘0011’	Right

3.2.2. Preprocessing and Feature Extraction

In order to extract the desired alpha brainwaves from the EEG signals, filtering was applied. More specifically, a second order IIR notch filter, having a quality factor Q equal to 35, was applied in order to remove mains frequency (50 Hz).

Consequently, the signals were further filtered by using a Butterworth IIR bandpass filter with cutoff frequencies of 5 and 15 Hz. The maximum loss in the passband was found to be equal to 0.1 dB. Similarly, the minimum attenuation in the stopband was measured to be equal to 30 db. The SciPy Python library was used for the design and application of the filters.

A typical sample of the signal filtering process performed is indicatively depicted in Figure 5. Specifically, the top graph shows the unfiltered signal acquired from the O1 position on the scalp of a subject, which gives the command for a ‘left’ movement of the robotic vehicle. As aforementioned in Table 1, the corresponding binary sequence is 1100 and this is why the signal amplitude is higher during the first half of the signal duration and lower during the last half. The middle graph of Figure 5 illustrates the signal filtered via the use of the notch filter while the bottom graph shows the signal further filtered with the bandpass filter.

Since alpha wave blocking is the reduction of alpha waves’ amplitude, this change can be measured by transforming the EEG signal from the time domain to the frequency domain. This is achieved by computing the Discrete Fourier Transform (DFT) of the signal using the FFT algorithm. The resulting amplitudes for the alpha wave frequency range are then summed. This process is repeated 4 times for each individual control signal; this is because control signals comprise of 4 2-second recording intervals.

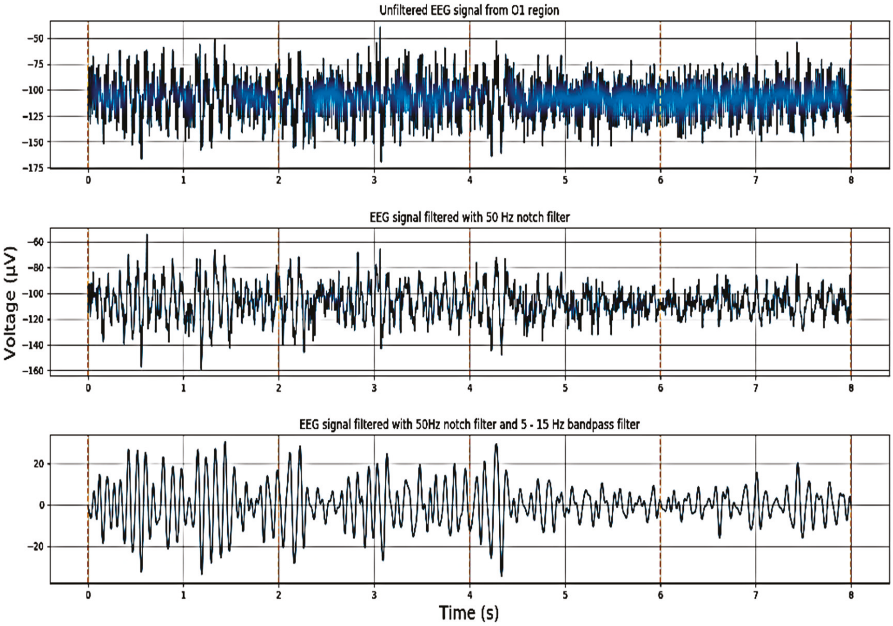


Figure 5. From top to bottom: Unfiltered electroencephalography (EEG) signal, EEG signal after being filtered with a 50 Hz notch filter, and final EEG signal with additional 5–15 Hz bandpass filter application.

Min-Max normalization is used to scale the features in the range of [0, 1], which are then saved as a dataset. The resulting feature vector consists of 8 amplitude sums, 4 for each channel (O1, O2). A total of 256 feature vectors are contained within the dataset. A visualization of an example feature vector for the movement "left" is depicted in Figure 6, where there are 8 different values, 2 for each bit. It is fairly easy to distinguish each individual bit value; in this case '1100'.

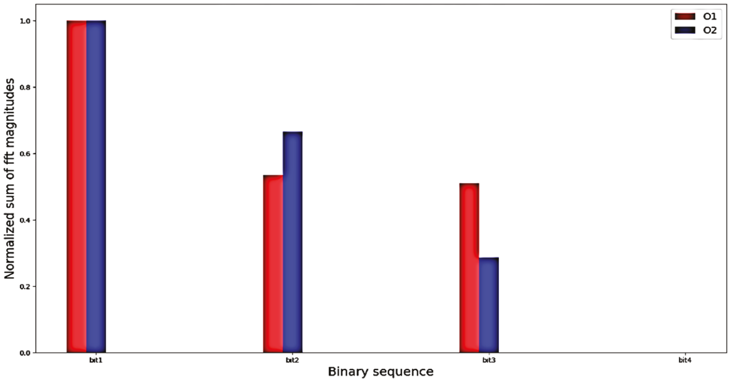


Figure 6. Bar chart showing the normalized sum of the FFT amplitudes for each EEG channel.

3.2.3. Classification and Translation

The classifier utilized for this research is a Multilayer Perceptron (MLP) neural network. This selection was made because MLP neural networks constitute a very popular machine learning technique and there is an abundance of successful applications of MLP neural networks in EEG signal classification and BCI research [27,28].

The classifier built consists of an input layer with 8 neurons, since the feature vector contains 8 amplitude sums, 4 for each channel. Furthermore, there are 4 neurons in the output layer because there are 4 available classes (forward, reverse, left, and right). Moreover, there are 2 hidden layers, each one consisting of 100 neurons.

The number of hidden layers and neurons was determined by a trial and error procedure. Specifically, 1–3 hidden layers were considered. In addition, for each layer the number of neurons examined was 20–200 with a step of 20. In total, 175 different network configurations were considered. It was concluded that a2 hidden layers network with 100 neurons in each layer achieved the desired performance in terms of classification accuracy. A graphical depiction of the classifier built is illustrated in Figure 7.

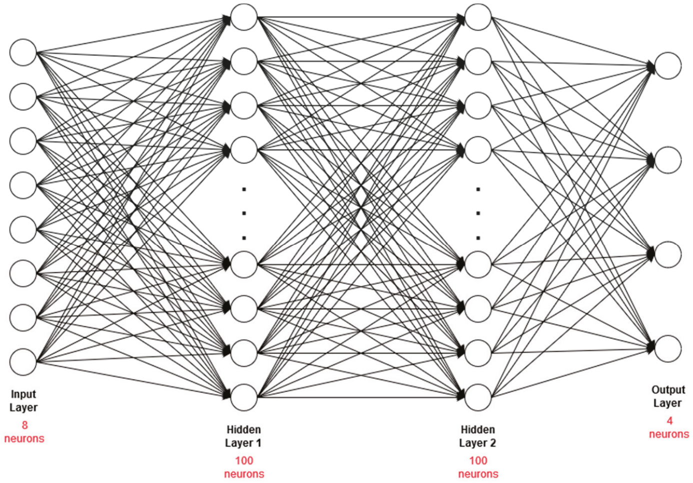


Figure 7. Structure of the neural network built.

The activation function for the hidden layers is the Rectified Linear Unit (ReLU). The advantages of ReLU include increased training speed and less suffering from the vanishing gradient problem [28]. The formula for ReLU is

$$ReLU(x) = max(0, x).$$

As for the output layers, the sigmoid function was used, which is given by the formula:

$$\sigma(x) = \frac{1}{1 + e^{-x}},$$

which bounds the output of each layer in the range of [0, 1]. This means that each neuron in the output layer produces probabilities of the input being one of the 4 commands. The command with the highest probability is selected.

The loss function used to measure the prediction error of the network during training is binary cross-entropy [29], which is widely used in binary classification problems. It is defined as

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} [y_n \cdot \log \hat{y}_n + (1 - y_n) \cdot \log(1 - \hat{y}_n)],$$

where N is the number of samples, y_n is the target output, and $\hat{y}_n$ is the predicted output. Finally, the optimization algorithm used to minimize the prediction error by adjusting the weight of each neuron is Adam, using the default hyperparameter values, as described in [30]. All models were trained in TensorFlow [31], using the Keras API [32].

In Figure 8 the neural network model training and validation loss is displayed. It can be distinguished that training could take place for a smaller number of epochs, since the loss is at an already acceptable value at around 25 epochs. The data used for validation is 40% of the total data.

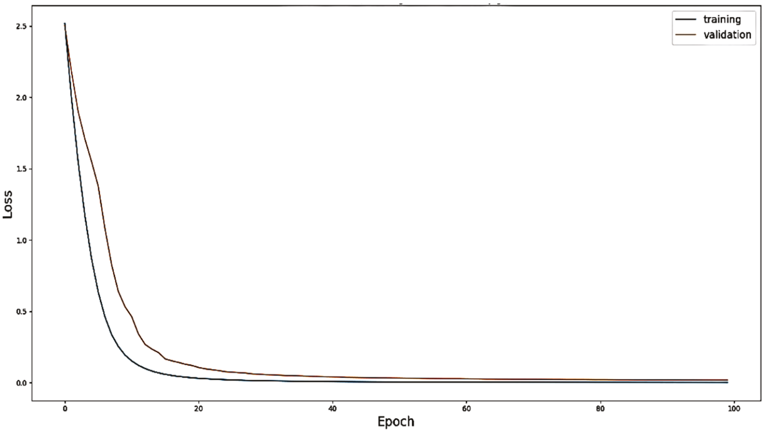


Figure 8. Training and validation model loss.

4. Results and Discussion

The performance of the developed system was evaluated by using both offline and online data which were gathered through a series of experimental tests performed in which 12 healthy subjects participated.

4.1. Evaluation with Offline Data

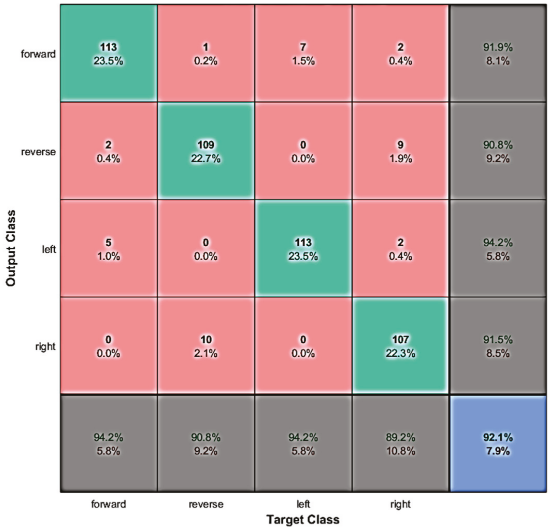
For the offline evaluation, the system was tested by using prerecorded data gathered from the same subjects used for recording the training data. Specifically, a small testing dataset of 50 feature vectors representing different movements was used. The neural network classified all of the movements correctly.

4.2. Real-Time Evaluation

After evaluating the system on offline data, a real-time performance analysis was carried out by using six female and six male subjects aged 20 to 28, and two female and two male subjects aged 32 to 40 years. The specific subjects were different from those that were used for the classifier training and offline evaluation. For this purpose, an experimental process was carried out. The subjects were instructed to move the robot in the following order: forward, reverse, left, and right consecutively.

Each one of the 12 subjects was briefed shortly on how the BCI works and how to issue each movement command to the robot. A small number of trial runs were performed for the subjects to get acquainted with the procedure. In total, 40 experimental tests were carried out. The total number of commands issued was 480.

The results of the experimental procedure showed that lowest classification accuracy achieved among the subjects was 85% while the highest one was 97.5%. The overall accuracy for all commands was 92.1%. The confusion matrix for the total number of commands considered for classification is illustrated in Figure 9, where green diagonal cells correspond to commands that are successfully classified, the red cells correspond to incorrectly classified commands, the gray column on the right displays the precision and false recovery rate of the classifier, the gray row in the bottom expresses the recall and the false negative rate of the classifier, and the blue cell displays the overall accuracy.

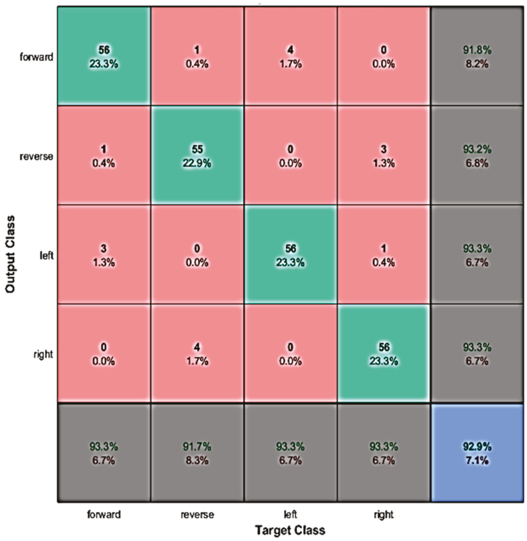

Output Class
Target Class

	forward	reverse	left	right	
forward	113 23.5%	1 0.2%	7 1.5%	2 0.4%	91.9% 8.1%
reverse	2 0.4%	109 22.7%	0 0.0%	9 1.9%	90.8% 9.2%
left	5 1.0%	0 0.0%	113 23.5%	2 0.4%	94.2% 5.8%
right	0 0.0%	10 2.1%	0 0.0%	107 22.3%	91.5% 8.5%
	94.2% 5.8%	90.8% 9.2%	94.2% 5.8%	89.2% 10.8%	92.1% 7.9%

Figure 9. Confusion matrix for all issued subject commands.

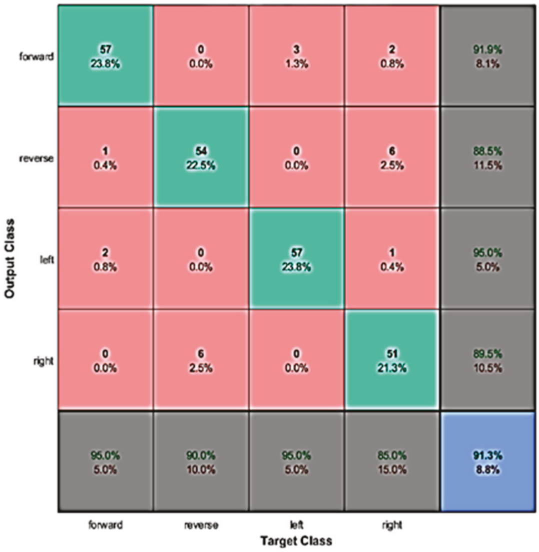
Next, for analysis purposes, the experimental results were studied according to the gender and the age of the subjects that participated in the experimental procedure.

Specifically, the results were first grouped and analyzed separately for each gender. The confusion matrices for the female subjects and the male subjects are depicted in Figures 10 and 11, respectively, where it is shown that the female subjects had a 1.6% higher classification accuracy compared to the male subjects (92.9% to 91.3%).


Output Class
Target Class

	forward	reverse	left	right	
forward	56 23.3%	1 0.4%	4 1.7%	0 0.0%	91.8% 8.2%
reverse	1 0.4%	55 22.9%	0 0.0%	3 1.3%	93.2% 6.8%
left	3 1.3%	0 0.0%	56 23.3%	1 0.4%	93.3% 6.7%
right	0 0.0%	4 1.7%	0 0.0%	56 23.3%	93.3% 6.7%
	93.3% 6.7%	91.7% 8.3%	93.3% 6.7%	93.3% 6.7%	92.9% 7.1%

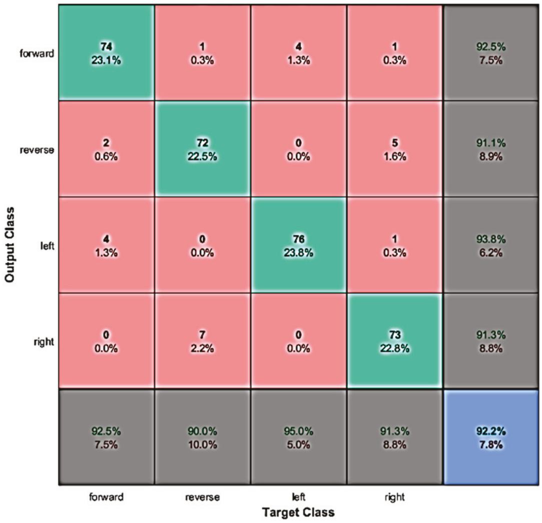
Figure 10. Confusion matrix for female subjects.



	forward	reverse	left	right	
forward	57 23.8%	0 0.0%	3 1.3%	2 0.8%	91.9% 8.1%
reverse	1 0.4%	54 22.5%	0 0.0%	6 2.5%	88.5% 11.5%
left	2 0.8%	0 0.0%	57 23.8%	1 0.4%	95.0% 5.0%
right	0 0.0%	6 2.5%	0 0.0%	51 21.3%	89.5% 10.5%
	95.0% 5.0%	90.0% 10.0%	95.0% 5.0%	85.0% 15.0%	91.3% 8.8%

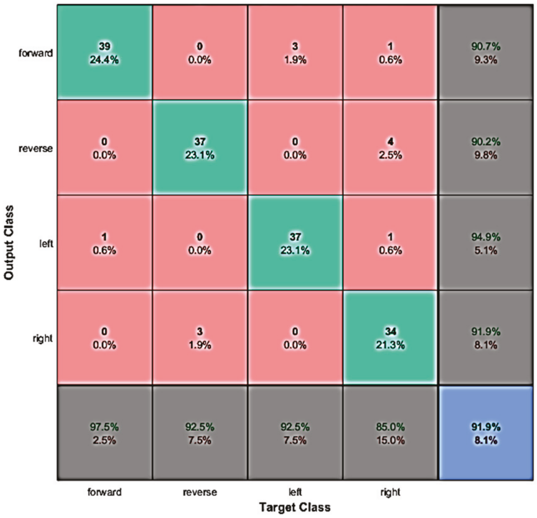
Figure 11. Confusion matrix for male subjects.

Next, the experimental results were grouped and analyzed according to the age of the subjects. The first group contains the results that refer to the eight subjects aged between 20 and 28 years and the second one the results derived by the four subjects aged between 32 and 40 years. The confusion matrices for the group 20–28 and the group 32–40 are depicted in Figures 12 and 13, respectively, where it is shown that these two groups have almost the same precision accuracy (92.2% for the subjects aged 20 to 28 and 91.9% for the subjects aged 32 to 40).



	forward	reverse	left	right	
forward	74 23.1%	1 0.3%	4 1.3%	1 0.3%	92.5% 7.5%
reverse	2 0.6%	72 22.5%	0 0.0%	5 1.6%	91.1% 8.9%
left	4 1.3%	0 0.0%	76 23.8%	1 0.3%	93.8% 6.2%
right	0 0.0%	7 2.2%	0 0.0%	73 22.8%	91.3% 8.8%
	92.5% 7.5%	90.0% 10.0%	95.0% 5.0%	91.3% 8.8%	92.2% 7.8%

Figure 12. Confusion matrix for ages 20 to 28.



	forward	reverse	left	right	
forward	39 24.4%	0 0.0%	3 1.9%	1 0.6%	90.7% 9.3%
reverse	0 0.0%	37 23.1%	0 0.0%	4 2.5%	90.2% 9.8%
left	1 0.6%	0 0.0%	37 23.1%	1 0.6%	94.9% 5.1%
right	0 0.0%	3 1.9%	0 0.0%	34 21.3%	91.9% 8.1%
	97.5% 2.5%	92.5% 7.5%	92.5% 7.5%	85.0% 15.0%	91.9% 8.1%

Figure 13. Confusion matrix for ages 32 to 40.

4.3. Discussion

The overall accuracy of 92.1% achieved by the proposed approach is considered to be rather satisfactory, especially given the fact that this rate is the result of real-time evaluation. It is also important to note that different subjects than the ones used for training were employed for this evaluation, a fact which attests to the robustness of the proposed method.

Better insight to the results can be gained by looking at the confusion matrix for all issued subject commands. It can be seen that the proposed approach not only achieves a satisfactory overall success rate, but also provides good performance per each individual movement.

Further analysis of inter-class performance shows that in 8.3% of the cases a 'reverse' command was issued, it was misclassified as a 'right' command. Moreover, the command 'left' was misclassified as a 'forward' command at a rate of 5.8% and the 'right' command as a 'reverse' command at a rate of 7.5%. This can be attributed to the fact that there is a short time delay until alpha wave amplitudes increase or decrease upon eye closing or opening, respectively. Therefore, these amplitudes are calculated into the next bit value, which can lead to errors.

A good indicator of the probability of a command being classified wrongly is the Hamming distance between each command (Table 2). Therefore, the 'forward' and 'reverse' commands are more likely to be misinterpreted into 'left' or 'right' commands and vice versa. Representing each command with more than four bits would increase the Hamming distance and, as a result, the system accuracy, but it would increase the overall recording time since the duration of every bit recording is two seconds.

Table 2. Hamming distances between robot commands.

Command		**'1010'**	**'0101'**	**'1100'**	**'0011'**
Forward	**'1010'**	0	4	2	2
Reverse	**'0101'**	4	0	2	2
Left	**'1100'**	2	2	0	4
Right	**'0011'**	2	2	4	0

The categorization of the experimental results performed according to the age of the subjects showed that the deviation in the classification accuracy of the age groups is negligible, probably because of the relatively small age difference between the two groups.

However, female subjects in the experimental procedure followed, achieved relatively higher classification accuracy than the male ones. This can be attributed to the fact that women in general exhibit greater alpha amplitudes than men [33,34].

On the other hand, although the performance of the proposed system was found to be successful, it is true that all the participants during the experiments made in this research work were healthy. Therefore, in real life conditions the effectiveness of experimental systems, like the one developed in this research work, is questionable because it strongly depends on the health conditions of their users who are supposed not only to be disabled persons but also having disability of various levels.

Moreover, the achievement of successful performance of a mobile robot within the territory of a controlled laboratory environment does not guarantee its effectiveness in real-world applications where the conditions are mostly variable and fuzzy.

Furthermore, the BCI systems that are based on a single signal may not be applicable to all users. Therefore, hybrid schemes which make combined use of various types of brain signals can be a more complex yet even more effective alternative.

5. Conclusions and Future Research

The research work, presented in this paper, concerns the development of a control system which guides the motion of a mobile robot via a synchronous and endogenous EEG-based BCI, which uses the alpha brain waveforms of a human operator.

Experiments made, with the involvement of 12 subjects who had minimum training, proved that the system developed is able to guide the robotic vehicle under control in forward, left, backward, and right direction according to the eyes' blinking of its human operator. The accuracy achieved ranges from 85% up to 97.5% among the subjects while the overall accuracy was found to be equal to 92.1% for all commands. Further analysis of the experimental data related with the classification accuracy between different genders and age groups showed that female subjects performed slightly better than male ones (92.9% to 91.3%, respectively), while there was just a trivial difference detected between subjects aged from 20 to 28 years and subjects aged from 32 to 40 years (92.2% to 91.9%, respectively).

Considering both the classification accuracy achieved, by applying real-time evaluation, and the robustness evinced by the fact that subjects involved during training were different than those during the experimental evaluation, it is concluded that the proposed method has the potential to be incorporated in applications such as the motion assistance to handicapped persons.

In the future, the conductors of this research work intend to experiment with hybrid BCIs where alpha brainwaves will be used along with brain signals of other type(s) such as P300 or SSVEP [35].

Moreover, task metrics, such as task completion time and path length traveled, and ergonomic metrics, such as mental workload of participants, can be additionally used for the accomplishment of multivariable evaluation of the performance of the system built [11].

Additionally, robot guidance can be assisted via additional sensors embedded into the robotic vehicle [36].

The detrimental effect of artifacts on EEG data can be removed by using modern algorithms that combine source decomposition with blind source separation and adaptive filtering [37].

Furthermore, enhanced performance can be achieved by applying advanced methods which have been proposed in order to add new knowledge to already learned models of robot semantic localization [38].

Author Contributions: All of the authors of this research article have extensively contributed to the work reported. Conceptualization, A.A., D.K., and N.K.; methodology, A.A. and D.K.; software, N.K.; validation, A.A., D.K., G.K., and N.K.; formal analysis, A.A., D.K., and N.K.; investigation, A.A., D.K., and N.K.; resources, D.K. and N.K.; data curation, A.A., D.K., and N.K.; writing—original draft preparation, A.A., D.K., and N.K.; writing—review and editing, A.A., D.K., G.K., and, N.K.; visualization, A.A. and N.K.; supervision, A.A., D.K., and G.K.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dixon, A.M.; Allstot, E.G.; Gangopadhyay, D.; Allstot, D.J. Compressed sensing system considerations for ECG and EMG wireless biosensors. *IEEE Trans. Biomed. Circuits Syst.* **2012**, *6*, 156–166. [CrossRef]
2. Perdiz, J.; Pires, G.; Nunes, U.J. Emotional state detection based on EMG and EOG biosignals: A short survey. In Proceedings of the 2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG), Coimbra, Portugal, 16–18 February 2017; pp. 1–4.
3. Valais, I.; Koulouras, G.; Fountos, G.; Michail, C.; Kandris, D.; Athinaios, S. Design and Construction of a Prototype ECG Simulator. *EJST* **2014**, *9*, 11–18.
4. Subha, D.P.; Joseph, P.K.; Acharya, R.; Lim, C.M. EEG signal analysis: A survey. *J. Med. Syst.* **2010**, *34*, 195–212. [CrossRef] [PubMed]
5. Wolpaw, J.R.; Birbaumer, N.; McFarland, D.J.; Pfurtscheller, G.; Vaughan, T.M. Brain–computer interfaces for communication and control. *Clin. Neurophysiol.* **2002**, *113*, 767–791. [CrossRef]
6. Abdulkader, S.N.; Atia, A.; Mostafa, M.S.M. Brain computer interfacing: Applications and challenges. *Egypt. Inform. J.* **2015**, *16*, 213–230. [CrossRef]
7. Katona, J.; Kovari, A. A Brain–Computer Interface Project Applied in Computer Engineering. *IEEE Trans. Educ.* **2016**, *59*, 319–326. [CrossRef]
8. Katona, J.; Kovari, A. The Evaluation of BCI and PEBL-Based Attention Tests. *Acta Polytechnica Hungarica* **2018**, *15*, 225–249.
9. Katona, J.; Kovari, A. Examining the learning efficiency by a brain-computer interface system. *Acta Polytechnica Hungarica* **2018**, *15*, 251–280.
10. Nicolas-Alonso, L.F.; Gomez-Gil, J. Brain computer interfaces, a review. *Sensors* **2012**, *12*, 1211–1279. [CrossRef]
11. Bi, L.; Fan, X.A.; Liu, Y. EEG-based brain-controlled mobile robots: A survey. *IEEE Trans. Hum. Mach. Syst.* **2013**, *43*, 161–176. [CrossRef]
12. Minguillon, J.; Lopez-Gordo, M.A.; Pelayo, F. Trends in EEG-BCI for daily-life: Requirements for artifact removal. *Biom. Signal Proces. Control* **2017**, *31*, 407–418. [CrossRef]
13. Padmavathi, R.; Ranganathan, V. A review on EEG based brain computer interface systems. *Int. J. Emerg. Technol. Adv. Eng.* **2014**, *4*, 683–696.
14. Gandhi, V. *Brain-Computer Interfacing for Assistive Robotics: Electroencephalograms, Recurrent Quantum Neural Networks and User-Centric Graphical Interfaces*, 1st ed.; Academic Press: London, UK, 2014; pp. 29–30.
15. Alexandridis, A.; Chondrodima, E.; Giannopoulos, N.; Sarimveis, H. A Fast and Efficient Method for Training Categorical Radial Basis Function Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2831–2836. [CrossRef] [PubMed]
16. Alexandridis, A.; Chondrodima, E.; Sarimveis, H. Radial Basis Function network training using a non-symmetric partition of the input space and Particle Swarm Optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 219–230. [CrossRef]
17. Lotte, F.; Bougrain, L.; Cichocki, A.; Clerc, M.; Congedo, M.; Rakotomamonjy, A.; Yger, F. A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update. *J. Neural Eng.* **2018**, *15*, 1–55. [CrossRef]
18. Tanaka, K.; Matsunaga, K.; Wang, H.O. Electroencephalogram based control of an electric wheelchair. *IEEE Trans. Robot.* **2005**, *21*, 762–766. [CrossRef]
19. Choi, K.; Cichocki, A. Control of a wheelchair by motor imagery in real time. In Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning, Daejeon, Korea, 2–5 November 2008; Springer: Berlin, Germany, 2008; pp. 330–337.
20. Ferreira, A.; Silva, R.L.; Celeste, W.C.; Bastos, T.F.; Filho, M.S. Human–machine interface based on muscular and brain signals applied to a robotic wheelchair. *J. Phys. Conf. Ser.* **2007**, *90*, 1–8. [CrossRef]
21. Mandel, C.; Luth, T.; Laue, T.; Röfer, T.; Graser, A.; Krieg-Bruckner, B. Navigating a smart wheelchair with a brain–computer interface interpreting steady-state visual evoked potentials. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 1118–1125.

22. Rebsamen, B.; Burdet, E.; Guan, C.; Zhang, H.; Teo, C.L.; Zeng, Q.; Ang, M.; Laugier, C. A brain controlled wheelchair based on P300 and path guidance. In Proceedings of the 1st IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 20–22 February 2006; pp. 1001–1006.
23. Iturrate, I.M.; Antelis, J.; Kubler, A.; Minguez, J. A noninvasive brain-actuated wheelchair based on a p300 neurophysiological protocol and automated navigation. *IEEE Trans. Robot.* **2009**, *25*, 614–627. [CrossRef]
24. Benevides, A.B.; Bastos, T.F.; Filho, M.S. Proposal of brain–computer interface architecture to command a robotic wheelchair. In Proceedings of the IEEE International Symposium in Industrial Electronics, Gdansk, Poland, 27–30 June 2011; pp. 2249–2254.
25. Samson, V.R.R.; Kitti, B.P.; Kumar, S.P.; Babu, D.S.; Monica, C. Electroencephalogram-Based OpenBCI Devices for Disabled People. In Proceedings of the 2nd International Conference on Micro-Electronics, Electromagnetics and Telecommunications, Visakhapatnam, India, 6–7 January 2017; pp. 229–238.
26. Olejarczyk, E.; Bogucki, P.; Sobieszek, A. The EEG split α peak: Phenomenological origins and methodological aspects of detection and evaluation. *Front. Neurosc.* **2017**, *11*, 506. [CrossRef]
27. Jana, G.C.; Swetapadma, A.; Pattnaik, P.K. Enhancing the performance of motor imagery classification to design a robust brain computer interface using feed forward back-propagation neural network. *Ain Shams Eng. J.* **2018**, *9*, 2871–2878. [CrossRef]
28. Subasi, A.; Erçelebi, E. Classification of EEG signals using neural network and logistic regression. *Comput. Methods Programs Biomed.* **2005**, *78*, 87–99. [CrossRef] [PubMed]
29. De Boer, P.T.; Kroese, D.P.; Mannor, S.; Rubinstein, R.Y. A tutorial on the cross-entropy method. *Ann. Oper. Res.* **2005**, *134*, 19–67. [CrossRef]
30. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
31. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
32. Chollet, F. Keras: The python deep learning library. *Astrophys. Source Code Libr.* **2018**. Available online: https://keras.io/k (accessed on 19 November 2019).
33. Wada, Y.; Takizawa, Y.; Zheng-Yan, J.; Yamaguchi, N. Gender differences in quantitative EEG at rest and during photic stimulation in normal young adults. *Clin. Electroencephalogr.* **1994**, *25*, 81–85. [CrossRef] [PubMed]
34. Corsi-Cabrera, M.; Ramos, J.; Guevara, M.A.; Arce, C.; Gutierrez, S. Gender Differences m in the Eeg During Cognitive Activity. *Int. J. Neurosci.* **1993**, *72*, 257–264. [CrossRef]
35. Amiri, S.; Fazel-Rezai, R.; Asadpour, V. A review of hybrid brain-computer interface systems. *Adv. Hum. Comput. Interact.* **2013**, *2013*, 1–12. [CrossRef]
36. Zantalis, F.; Koulouras, G.; Karabetsos, S.; Kandris, D. A Review of Machine Learning and IoT in Smart Transportation. *Future Internet* **2019**, *11*, 94. [CrossRef]
37. Jafarifarmand, A.; Badamchizadeh, M.A. EEG Artifacts Handling in a Real Practical Brain-Computer Interface Controlled Vehicle. *IEEE Trans. Neural Syst. Rehabilit. Eng.* **2019**, *27*, 2000–2008. [CrossRef]
38. Cruz, E.; Rangel, J.C.; Gomez-Donoso, F.; Bauer, Z.; Cazorla, M.; García-Rodríguez, J. Finding the place: How to train and use convolutional neural networks for a dynamically learning robot. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.

electronics

MDPI

Article

Fallen People Detection Capabilities Using Assistive Robot

Saturnino Maldonado-Bascón, Cristian Iglesias-Iglesias, Pilar Martín-Martín * and Sergio Lafuente-Arroyo

Signal Theory and Communications Department, Alcalá University, Alcalá de Henares, 28805 Madrid, Spain
* Correspondence: p.martin@uah.es; Tel.: +34-918-856-735

Received: 24 July 2019; Accepted: 18 August 2019; Published: 21 August 2019

Abstract: One of the main problems in the elderly population and for people with functional disabilities is falling when they are not supervised. Therefore, there is a need for monitoring systems with fall detection functionality. Mobile robots are a good solution for keeping the person in sight when compared to static-view sensors. Mobile-patrol robots can be used for a group of people and systems are less intrusive than ones based on mobile robots. In this paper, we propose a novel vision-based solution for fall detection based on a mobile-patrol robot that can correct its position in case of doubt. The overall approach can be formulated as an end-to-end solution based on two stages: person detection and fall classification. Deep learning-based computer vision is used for person detection and fall classification is done by using a learning-based Support Vector Machine (SVM) classifier. This approach mainly fulfills the following design requirements—simple to apply, adaptable, high performance, independent of person size, clothes, or the environment, low cost and real-time computing. Important to highlight is the ability to distinguish between a simple resting position and a real fall scene. One of the main contributions of this paper is the input feature vector to the SVM-based classifier. We evaluated the robustness of the approach using a realistic public dataset proposed in this paper called the Fallen Person Dataset (FPDS), with 2062 images and 1072 falls. The results obtained from different experiments indicate that the system has a high success rate in fall classification (precision of 100% and recall of 99.74%). Training the algorithm using our Fallen Person Dataset (FPDS) and testing it with other datasets showed that the algorithm is independent of the camera setup.

Keywords: assistive robot; fall detection; lying-pose recognition; deep learning; mobile robot; convolutional neural network; support vector machine

1. Introduction

Falls are considered one of the most serious issues for the elderly population [1]. In general, those falls cause injury, loss of mobility, fear of falling and even death. Some studies suggest that falls where the patient has been waiting a long time on the ground before help arrives are associated with bigger health problems [2]. Reliable fall detection systems are an essential research topic for monitoring the elderly and people with disabilities who are living alone [3].

Many approaches have been proposed using many different kinds of devices and methodologies and some of them are summarized by Noury [4], Mubashir [5], Igual [6] and Khan [7]. Principally, all proposed approaches can mostly be divided into two big groups—wearable-based and vision-based devices methods.

Studies based on wearable devices are growing fast and they rely on sensors that are attached to the person's body as accelerometers, gyroscopes, interface pressure sensors and magnetometers [8–11]. Although these approaches have provided high detection rates by using small and cheap technology, they require active cooperation by wearing the sensors. As a consequence, for long-term use, they are not a practical solution by themselves.

On the contrary, vision-based devices do not require any support from the elderly. At the same time, cameras nowadays are increasingly used in our daily lives. Vision-based fall detection systems analyze in real-time the position and shape of the person using different kinds of algorithms that combine standard computing platforms and low-cost cameras. Compared with other methods, the vision-based methods promise results due to the fast advances in computer-vision and video-camera technologies, such as the economical Microsoft Kinect [12–14]. The combination of video-based and ambient sensor-based systems (external sensors embedded in the environment, such as infrared, pressure and acoustic sensors [15]) also provide excellent results.

Mobile robots are the right solution for keeping a single person in view when compared to static cameras [14,16,17]. To avoid terrain difficulty, Máthé et al. [18] and Iuga et al. [19] proposed methods that use uncrewed aerial vehicles (UAVs) as mobile robots. A useful aspect of patrol robots, instead of robots that keep the person continuously in view, is the integration of privacy protection and real-time algorithms. As the person is not under supervision all the time, especially in particular locations like the bathroom, the elderly feel more relaxed because their privacy is less invaded.

In this work, we deal with the fall detection problem in the case of having one, two, or more people in the same environment. We used our multifunctional and low-cost mobile robot equipped with a 2D image-based fall detection algorithm as a patrol robot. The assistive robot autonomously patrols an indoor environment, and when it detects falls, it activates an alarm. The system was designed to recognize lying-poses in single images without any knowledge about the background. Additionally, the robot relocates itself in case of doubt in detection. However, we assume that it is improbable that a patrol robot takes an image during the falling; therefore, this work focuses on detecting falls in a short interval after the event of falling.

Additionally, to analyze the effectiveness of the approach, we provide a new dataset to be used in fall detection algorithms. The main features of this dataset are:

- several scenarios with variable light conditions,
- different person sizes,
- images with more than one actor,
- persons wearing different clothes,
- several lying-position perspectives and
- resting and fallen persons.

The remainder of the paper is organized as follows. Section 2 describes the needs of and challenges for fall detection systems and reviews the work related to fall detection vision-based approaches. Section 3 describes the design and methodology of the proposed fall detection method in detail. We describe the system architecture in Section 3.1. Section 3.2 focuses on person detection and fall classification is analyzed in Section 3.3. In Section 4, a new dataset is described and the method is evaluated. Section 4.1 describes the Fallen Person Dataset (FPDS) in detail. Section 4.2 presents the used metrics for measuring the effectiveness of the technique. The following three Sections 4.3–4.5, outline the carried-out experiments to evaluate the proposed approach from different points of view. Two evaluations of the method, relocation of the patrol robot and performance verification over other datasets were done and they are outlined in the last two Sections 4.6 and 4.7. Finally, in Section 5, conclusions and future research directions are identified.

2. Vision-Based System Overview

Vision-based systems offer many advantages over wearable sensor-based systems. Mainly, they are more robust and once they are installed, the person can forget about them. In these systems, cameras play an important role. If we consider the number and type of cameras, there are mainly three groups [20]—single camera, multicamera and depth cameras. For 2D-vision systems, only one uncalibrated camera is required but for 3D vision systems, we need a calibrated single camera or multicamera.

The most extensive systems are based on a single camera due to their simplicity and price. Particularly in the case of fixed cameras, since cameras are static, and background subtraction can mainly be applied to find the person in the image [21]. Kim et al. [22] proposed one of the more used real-time foreground-background methods. However, the person could be integrated into the background when they have been sitting on a couch for long. Several approaches show that it is possible to achieve good results using a single camera. Charfi et al. [23] proposed a technique based on feature extraction, an SVM-based classifier and a final decision step. Liu et al. [24] used a k-nearest neighbor classifier and Wang [25] performed multi-viewpoint pose parsing based on part-based detection results.

Fixed cameras are efficient only if the camera is placed in the ceiling of the room to avoid occlusion objects. However, the camera does not have good access to the vertical parameter of the body, which provides essential information for fall detection [26]. Another intelligent solution consists of using an assistive robot equipped with a single camera. In that case, occlusion or doubtful cases can be solved using different viewpoints that can be taken from the moving robot.

On the other hand, a good solution for solving the problem of occlusion would use a system with multiple cameras. However, the main issues in those cases are time-consuming calibration to compute reliable 3D information and the synchronization process between the different cameras. Some studies have been working out these problems, such as Rougier et al. who, in Reference [27], proposed a method based on Gaussian Mixture Model (GMM) classification and human-shape deformation for uncalibrated single- and multicamera systems.

Depth cameras, such as Kinect, provide several advantages, for example, independence from light conditions, silhouette ambiguity of the human body, simplification of background-subtraction tasks and reduction of the time needed for calibration [12,13].

In general, vision-based fall detectors have some challenges to resolve for good performance in the different situations that the person can be found:

- high variability of possible body orientations on the floor,
- different person sizes,
- wide range of background structures and scenarios and
- occlusions being frequent cases in the fall detection context.

Based on all previously mentioned reasons, our proposal is a vision-based learning solution for fall detection by using a single RGB camera mounted on an assistive patrol robot. The robot patrols around the indoor environment and, in case of fall detection, activates an alarm. The proposed method deals with three of the previous four points, as is shown in the Experiment Results section. How to improve our work with the occlusions is further investigated.

3. Proposed Fall Detection Approach

Our approach solves the fall detection problem in an end-to-end solution based on two steps—person detection and fall classification. The person detection algorithm aims to localize all persons in an image. Its output is the enclosing bounding boxes and the confidence scores that reflect how likely it is that the boxes contain a person. Fall classification estimates if the detected person is in a fall or not.

In this approach, we propose to combine the YOLOv3 algorithm based on a Convolutional Neural Network (CNN) for person detection and a Support Vector Machine (SVM) for fall classification. The main steps of our detection system (Figure 1) are as follows:

- Take a single image.
- Person detection. Results are the coordinates of the bounding box of the detected human body.

- Feature extraction from the bounding box coordinates.
- Fall identification.
  - Nonfall detection—continue taking new images.
  - Fall detection—ask for confirmation of the fall.
  - Doubt detection—the bounding box is too small, too big, or is located at the edges of the image. The robot needs to relocate itself to center the possible fall detection with the proper dimensions.

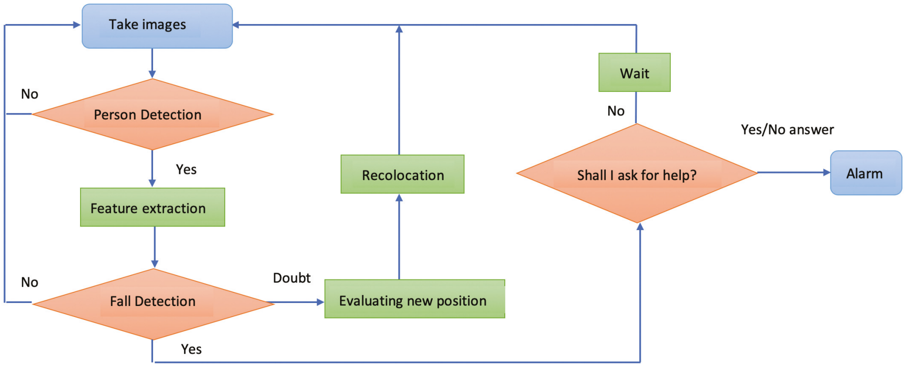


Figure 1. Flowchart of fall detection approach.

3.1. System Architecture

As a base for the fall detection approach, we used the assistance robot LOLA, designed entirely by our research team to monitor and help the elderly and any other person with a functional disability who lives alone. The main idea behind the LOLA robot was to be an assistive robot that could also work as a rollator for helping to walk or as a table to transport objects due to its shape—80 cm height, 58 cm width and 70 cm depth (Figure 2).

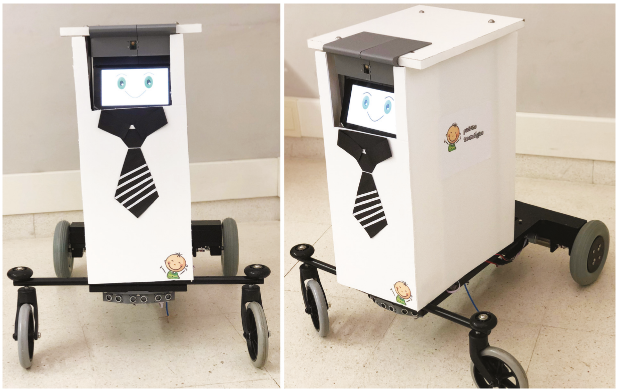


Figure 2. LOLA assistive robot.

The system is equipped with an Arduino Mega board, various sensors, a single RGB camera and Raspberry V3 B+. We also needed a connection to a server to perform the heavy workload—image processing and the fall detection algorithm. This connection could be WIFI to a remote server or ethernet to a laptop located in the robot. The camera is located 76 cm above the floor and takes images of 640 × 480 pixels (Figure 3).

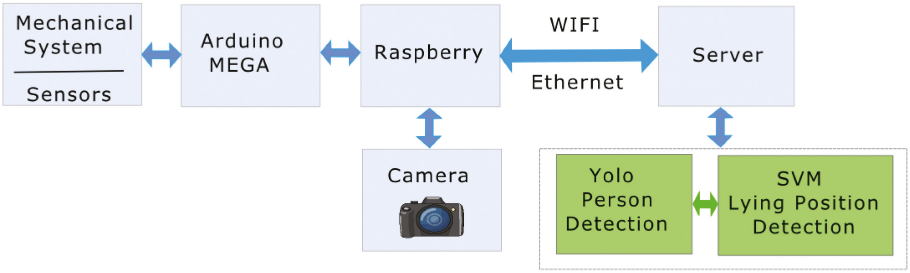


Figure 3. System-architecture overview.

3.2. Deep Learning-Based Person Detection

CNNs are one of the most popular machine-learning algorithm types at present and it has been decisively proven over time that they outperform other algorithms in accuracy and speed for object detection [28].

Algorithms for object detection using CNN can be broadly categorized into two-stage and single-stage methods. The two-stage algorithm based on classification first generates many proposals or interesting regions from the image (body) and then those regions are classified using the CNN (head). In other words, the network does not check the complete image; instead, it only checks parts of the image with a high probability of containing an object. Region-CNN (R-CNN) proposed by Ross Girshick in 2014 [29] was the first of this series of algorithms that was later modified and improved, for example, fast R-CNN [30], faster R-CNN [31], R-FCN [32], Mask R-CNN [33] and Light-Head R-CC [34]. However, single-stage algorithms based on regression do not use regions to localize the object within the image; the predict bounding boxes and class probabilities at the whole image. The most known examples of this type of algorithm are Single Shot Detector (SSD), proposed by Liu et al. [35] and 'you only look once' (YOLO) proposed by Joesph Redmon et al. in 2016 [36]. YOLO has been updated to versions YOLOv2, YOLO9000 [37] and YOLOv3 [38]. In this paper, we decide to apply real-time object detection system YOLOv3 for person detection, which has proven to be an excellent competitor to other algorithms in terms of speed and accuracy.

The YOLO network takes an image and divides it into S × S grids. Each grid predicts B bounding boxes $\{bi\}, i = 1, \ldots, B$ and provides a confidence score for each of them $Conf_{bi}$, which reflects how likely the box contains an object. Bounding boxes with this parameter above a threshold value are selected and used to locate the object, a person in our case. The bounding box position is the output of this stage for our algorithm.

3.3. Learning-Based Fall/Nonfall Classification

The effectiveness of SVM-based approaches for classification has been widely tested [39–41]. The SVM algorithm defines a hyperplane or decision boundary to separate different classes and maximize the margin (maximum distance between data points of the classes). Support vectors are training data points that define the decision boundary [42]. To find the hyperplane, a constrained minimization problem has to be solved. Optimization techniques such as the Lagrange multiplier method are needed.

In the case of nonlinearly separable data, data points from initial space R_d are mapped into a higher dimensional space Q where it is possible to find a hyperplane to separate the points. With this, the classification-decision function becomes

$$f(x) = sgn(\sum_{i=1}^{N_s} y_i \alpha_i K(x, s_i) + b) \tag{1}$$

where training data are represented by $\{x_i, y_i\}$, $i = 1, \ldots, N$, $y_i \in \{-1, 1\}$, b is the bias, α_i, $i = 1, \ldots, N$ are the Lagrange multipliers obtained during the optimization process [43] and s_i, $i = 1, \ldots, N_s$ are the support vectors, for which $\alpha_i \neq 0$ and $K(x, x_i)$ is a kernel function. A Radial Basis Function (RBF) was used as a kernel in this study:

$$K(x, x_i) = e^{-\gamma ||(x - x_i)||^2} \tag{2}$$

where γ is the parameter controlling the width of the Gaussian kernel.

The accuracy of the SVM classifier depends on regularization parameter C and γ. C is the parameter that controls the penalization associated with the training samples that are misclassified and γ defines how far the influence of a single training point reaches. So, both parameters must be optimized for every different task in particular, for example, by using cross-validation.

The selection of the right features or input parameters to the SVM plays an important role in having a high-performance classification algorithm. Some features are most widely used in the literature as aspect ratio (AR), change in AR (CAR), fall angle (FA), center speed (CS) or head speed (HS) [21,44,45]. However, after analyzing the parameters that provide the best trade-off performance for goals to achieve in our approach, using the bounding box data of a detected person, we defined the input feature vector for the SVM classifier as

- Aspect ratio of bounding box, AR_i:

$$AR_i = \frac{W_{bi}}{H_{bi}} \tag{3}$$

- Normalized bounding box width, NW_i:

$$NW_i = \frac{W_{bi}}{W_{image}} \tag{4}$$

- Normalized bounding box bottom coordinate, NB_i:

$$NB_i = 1 - \frac{Ydown_{bi}}{H_{image}} \tag{5}$$

where W_{bi} = $Xright_{bi} - Xleft_{bi}$, H_{bi} = $Ydown_{bi} - Xtop_{bi}$ are the width and height of bounding box $\{bi\}$, respectively, calculated from the bounding box position provided by YOLOv3 $\{Xleft_{bi}$, $Xright_{bi}$, $Ytop_{bi}$, $Ydown_{bi}\}$ and W_{imagen}, H_{imagen} are the width and height of the overall image. Point $(0, 0)$ is at the top-left corner of the overall image. Parameter NB_i defines the distance from the bottom of the image to the lower part of the normalized bounding box. As the values of the NB_i and NW_i parameters are between 0 and 1, in order to give a similar weight to AR_i, we needed to adjust its value as input to the SVM. We analyzed the data and W_{bi} was lower than $10H_{bi}$ for all cases, so we normalized AR_i by 10 in order to get a feature in [0,1]. Therefore, we considered detection if $W_{bi} < 10H_{bi}$.

Parameter AR_i is the most significant feature that characterizes the fall. As can be seen from the examples in Figure 4a,b, a person standing upright has a small AR_i, while this ratio is large in the case of a person lying in a horizontal body orientation position. However, this parameter alone is not enough. There are some cases where the person is in a lying-position but this parameter does not show it; this is the case of lying in a vertical body orientation position, as we show in Figure 4c.

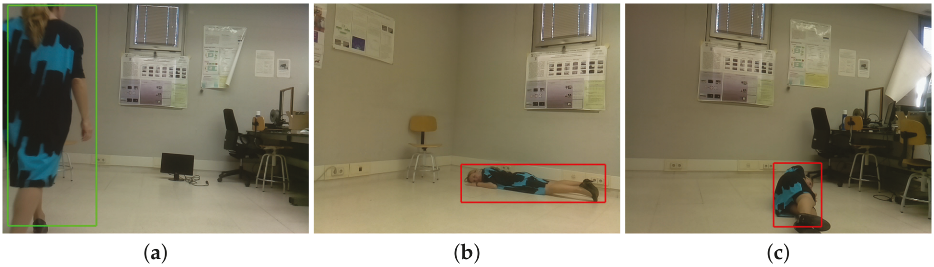


Figure 4. Aspect ratio. (**a**) Standing person $AR_i = 0.402$. (**b**) Fallen person in horizontal-pose orientation position $AR_i = 3.810$. (**c**) Fallen person in vertical-pose orientation position $AR_i = 0.751$.

One of the main goals of the algorithm is the ability to differentiate between fallen people and resting situations. Figure 5 shows one example of how the optical perspective in the cameras works. The object size in the image (in pixels) depends on the real image size (in mm) and the distance from the camera to the object [46]:

- Objects with the same size at different distances from the camera (object planes) appear with a different size (pixels) in the image plane; the closest one is visible in a larger size (Figure 5a);
- objects with the same size at the same distance to the camera (object planes) appear with the same size (pixels) in the image plane (Figure 5b). If objects are at different heights in the object plane, the same happens in the image plane.

When we compare a fallen person and a resting person at the same distance from the camera, the situation is the one observed in Figure 5b. The resting person is the person in the higher position. As shown in Figure 6a, the AR_i and NW_i parameters in both cases were the same (same size of bounding box); however, the NB_i parameter was different NB_1, NB_2. For the same value, NB_1, the bounding box size for a fallen person should be the red one (see Figure 6b).

Therefore, proposed parameters AR_i, NW_i and NB_i provide needed information for differentiating those situations and, during the training stage, the SVM learns the relation between them in both cases (fall and resting position).

Figure 7 shows the previous explanation with real images. It contains three pairs of images where fallen and resting persons are at the same distance from the camera (1.5, 2 and 3 m away). Table 1 shows the parameters provided to the SVM in those situations. As can be seen in the table, each pair of images have a similar NW_i parameter (slight differences are due to not being precisely at the same position from the camera). However, parameter NB_i had a larger value in the nonfall situation because the body was in a higher position in the image.

Table 1. Input parameters to the support vector machine (SVM) from images in Figure 7.

1.5 m			**2 m**			**3 m**		
AR$_i$	**NW$_i$**	**NB$_i$**	**AR$_i$**	**NW$_i$**	***NB$_i$***	**AR$_i$**	**NW$_i$**	**NB$_i$**
5.33	0.85	0.014	4.54	0.61	0.11	3.47	0.39	0.22
4.64	0.69	0.25	4.41	0.56	0.30	3.06	0.37	0.32

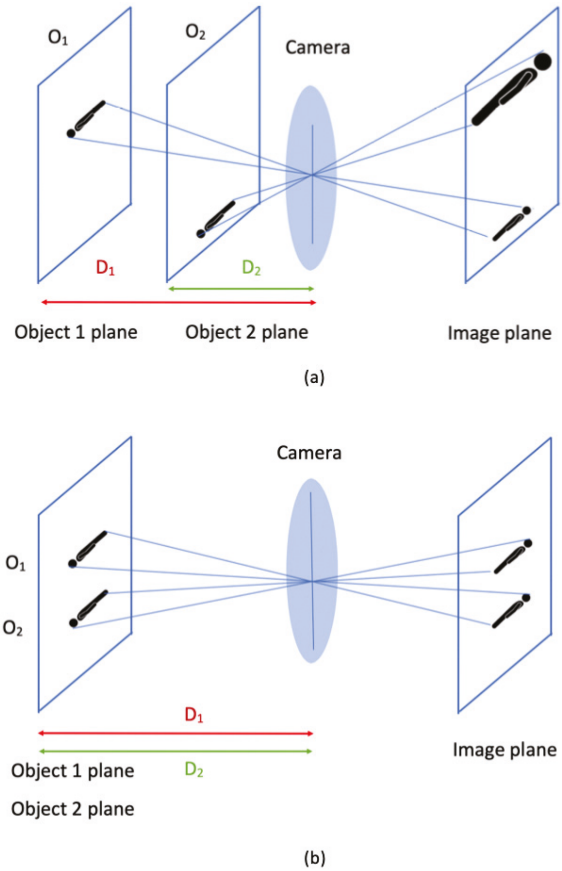


Figure 5. Optical perspective. Image plane for same object at (**a**) different distances and (**b**) different heights.

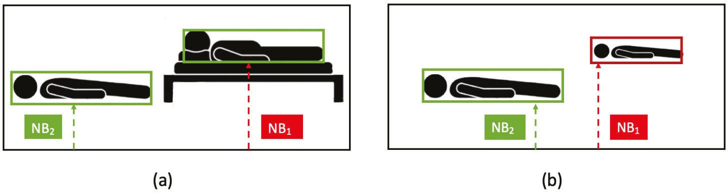


Figure 6. Relation between the NB_i parameter and the bounding box size. (**a**) Fallen and resting persons at same distance from camera. (**b**) Two fallen persons at different distances from camera.

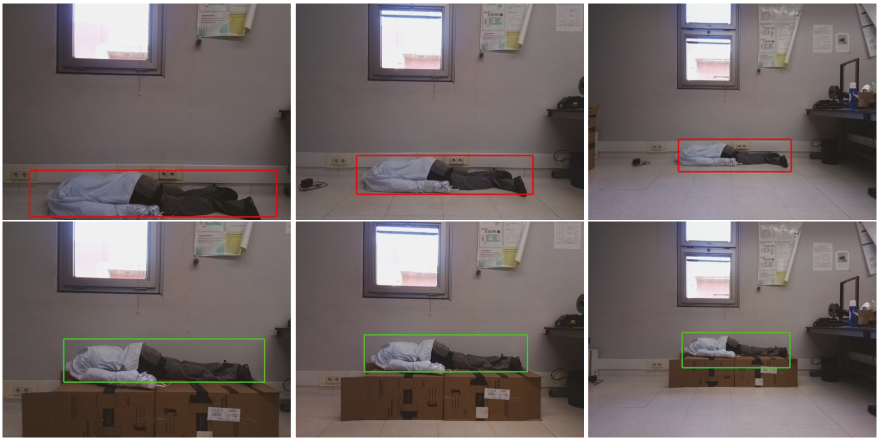


Figure 7. Fall/nonfall detection 1.5, 2 and 3 m away (each pair of images are in the same column).

4. Experiment Results

4.1. FPDS Dataset

Analysis and comparison of different fall detection algorithms is a real problem due to the lack of public datasets with a large number of people in lying-positions [21,47,48]. ImageNet [28] and MS-COCO [49] are examples of those large image datasets. Some fall detection datasets provide images or videos with the camera situated in different positions but most of them in simulated environments [23,48,50–52]. However, they are neither large enough nor have all the required image variations for testing our experiments—several environments, more than one person in each image, persons in resting positions, falls with a variety of body orientations and persons with different sizes and clothes.

For all those reasons, in this paper, we present our own dataset (FPDS) to be used in fall detection algorithms. All images were taken by using a single camera inserted in a robot at 76 cm above the floor. This dataset consisted of a total of 2062 manually labeled images with 1072 falls and 1262 people standing up, sitting in a chair, lying on the sofa, walking and so forth. Images could have more than one actor and were recorded from different perspectives (Figure 8). An essential feature of this dataset compared with other datasets was having actors with a height range of 1.2–1.8 m (see Figure 9).

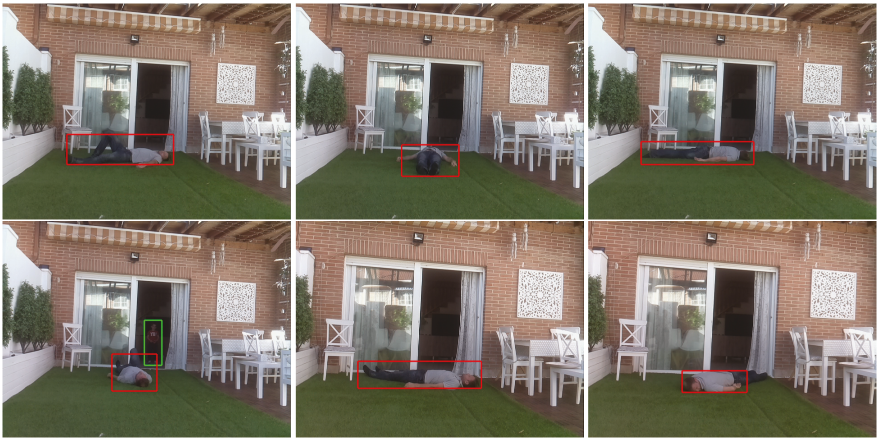


Figure 8. Fallen Person Dataset (FPDS) images with different lying-body orientations.

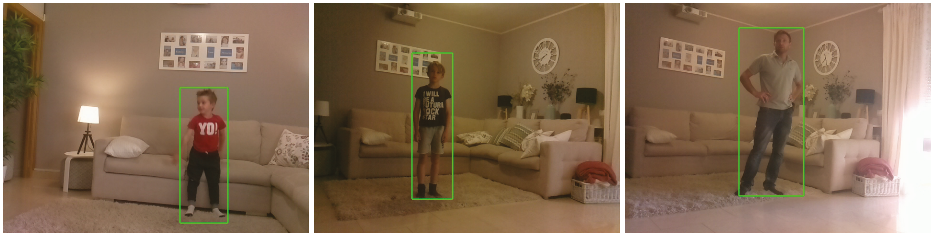


Figure 9. FPDS images with different person sizes—1.2, 1.4 and 1.8 m height.

Images were taken in eight different environments with variable illumination, as well as shadows and reflections, defining eight splits. Figure 10 and Table 2 show sample images and the characteristics of the FPDS, respectively.

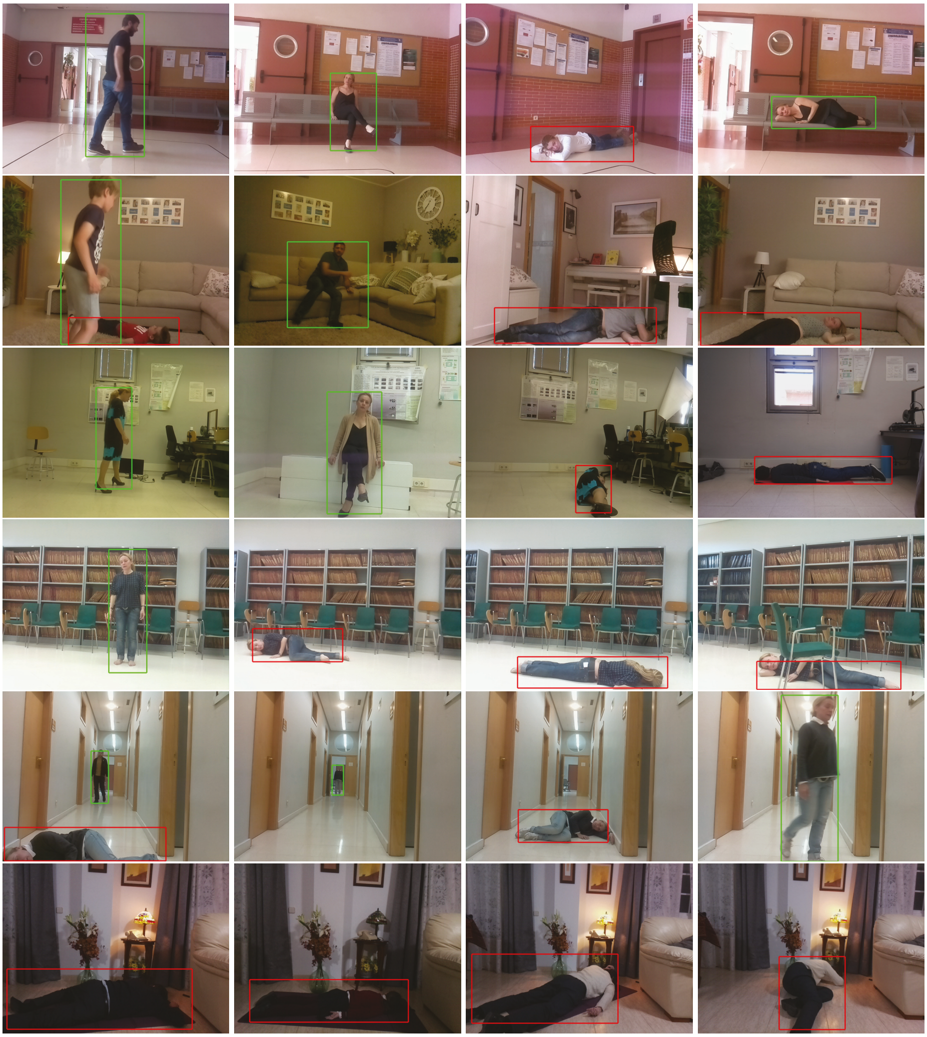


Figure 10. *Cont.*

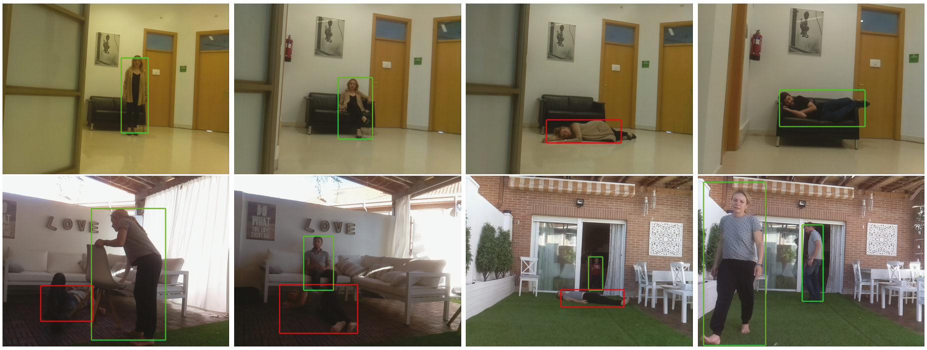


Figure 10. Ground-truth image examples of the FPDS. Each row belongs to a different split. Bounding boxes are red/green in case of fall/nonfall detection.

Table 2. FPDS dataset characteristics.

	Split 1	**Split 2**	**Split 3**	**Split 4**	**Split 5**	**Split 6**	**Split 7**	**Split 8**	**Total**
Number of falls	278	223	180	104	49	42	15	181	1072
Number of nonfalls	175	82	175	3	704	0	39	84	1262
Number of images	400	323	368	117	553	42	51	210	2064

FPDS dataset consists of images and txt files with the same name. These files contain five parameters per bounding box in the image— bounding box coordinates $\{Xleft_{bi}, Xright_{bi}, Ytop_{bi}, Ydown_{bi}\}$ and the classification label y ($y = 1$ fall, $y = -1$ nonfall). Additionally, in the dataset we provided some sample images of a well-defined pattern (chessboard) taken with the camera from different perspectives for calibration purposes. FPDS dataset is public and available at http://agamenon.tsc.uah.es/Investigacion/gram/papers/fall_detection/FPDS_dataset.zip.

For all experiments, training and testing images belonged to different splits to correctly evaluate the ability of the algorithm to learn. We built training set L with splits 1, 2 and 3, by using 681 falls and 432 nonfalls in a total number of 1084 images. Testing set T was built by using from 4 to 8 splits, with 391 falls and 830 nonfalls in a total number of 973 images.

4.2. Metrics

To investigate the effectiveness of the method, we evaluate fall detection at the classifier-output level by measuring error rates, computed from good and misclassified images. However, to fully evaluate the algorithm, we needed to measure the precision and recall parameters [23]. Precision provides information about the proportion of positive fall identifications that are actually falls and recall the proportion of falls that were identified correctly. Unfortunately, these parameters work in different directions, meaning that improving precision typically reduces recall and vice versa:

$$Pr = \frac{TP}{TP + FP} \tag{6}$$

$$Re = \frac{TP}{TP + FN} \tag{7}$$

being

- True positives (TP)—number of falls correctly detected,
- false negatives (FN)—number of falls not detected and
- false positives (FP)—number of nonfalls detected as falls.

4.3. Experiment 1: Fall Classification

In this first experiment, we evaluated the performance of the learning-based fall/nonfall classification algorithm by itself without considering the person detection part. We use the ground-truth hand-labeled bounding boxes from our dataset as inputs. Cross-validation was performed on the training set to find the optimal C and γ values in the RBF SVM classifier. Figure 11 shows the accuracy-level curves for both parameters. We selected $\gamma = 2$ and $C = 128$ with an accuracy of 99.55%. These values were also established for Experiments 2 and 3.

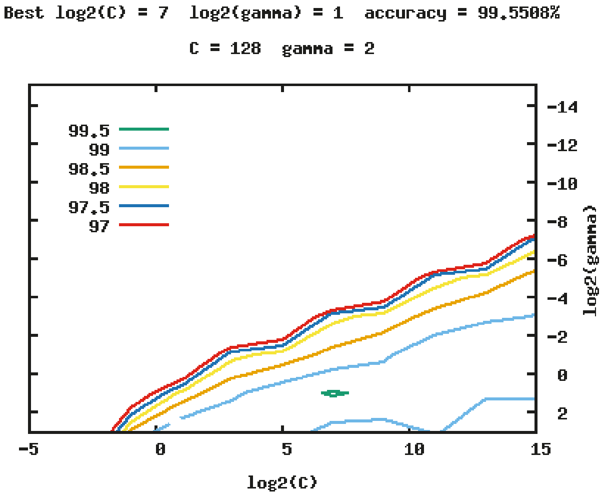


Figure 11. Accuracy-level curves during cross-validation for γ and C parameters in Experiment 1.

We summarize the experiment results in only one table to help with comparisons (Table 3). The first row of this table are the results of this experiment. The fall classifier detected 390 true positives, 1 false negative and 0 false positives, which means precision and recall of 100% and 99.74%, respectively. These results confirm a great selection of the selected input parameters to the SVM classifier.

Table 3. Performance over testing set T in the FPDS dataset.

	TP	*FN*	*FP*	*Pr* (%)	*Re* (%)
Experiment 1: Fall classification	390	1	0	100	99.74
Experiment 2: Fall detection algorithm	304	87	9	97.12	77.74
Experiment 3: Fall detection with pose correction	360	31	17	95.49	92.07

Several approaches have been proposed to detect falls, with good results. However, only a few of them take into account realistic datasets with different normal daily situations. One of the more complicated situations to solve is not detecting falls versus standing but rather falls versus resting situations where the person has a similar pose orientation. Our fall classifier can detect both situations in all cases that were tested. Figure 12 shows two examples.

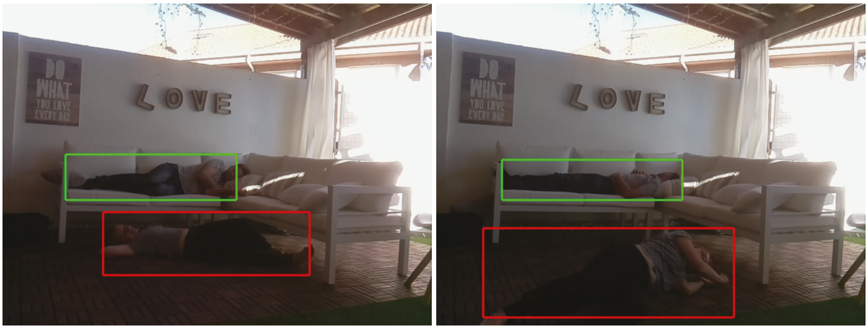


Figure 12. Example images from testing test where algorithm differentiates between fallen person and resting person.

4.4. Experiment 2: Fall Detection Algorithm

The next experiment evaluated the performance of the overall end-to-end fall detection algorithm—person detection and fall classification. In this case, the person detection part was done by using deep learning method YOLOv3.

To maximize performance, the confidence score of bounding box conf$_{bi}$ provided by YOLOv3 should have been above than a certain threshold Conf$_i$. We selected threshold value Conf$_i$ = 0.2 for having good trade-off performance between recall and precision. Figure 13a shows this point by '*'.

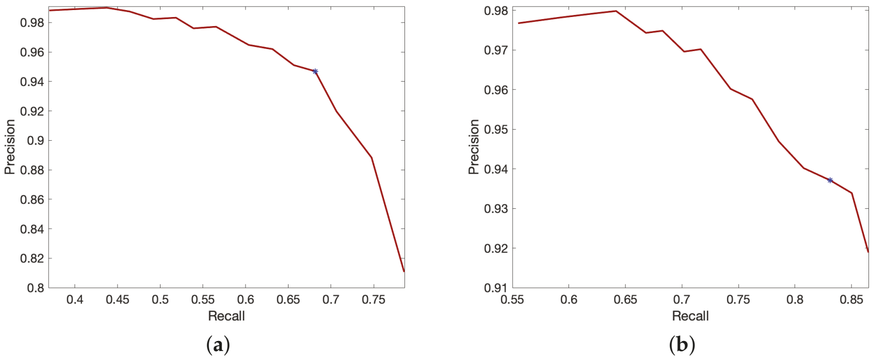


Figure 13. Recall and precision metrics for different thresholds. (**a**) Experiment 2, conf$_i$. (**b**) Experiment 3, conf$_r$.

Note the terminology—subindex "*i*" is used for parameters assigned to the "image directly from the camera" to differentiate them from the parameters assigned to the "rotated images" with subindex "*r*" that is explained in the next subsection.

We used Intersection over Union (IoU) as an evaluation metric to compare the bounding boxes provided by the fall detection algorithm and the ground-truth hand-labeled images from our dataset. To set a threshold value for the IoU, called IoU$_i$, we analyzed how this value affects the precision and recall parameters. It was observed that the values of these metrics were almost independent of the selected threshold, setting; in that case, value to IoU$_i$ = 0.2. Values Conf$_i$ = 0.2 and IoU$_i$ = 0.2 were also established in Experiment 3.

As in the preceding subsection, the second row of Table 3 shows the results of testing set *T* for this experiment. It detected 304 true positives, 87 false negatives and 9 false positives. The values of precision and recall, in this case, were 97.12% and 77.74%, respectively. The false alarms were mainly caused by errors in the person detection step of the overall algorithm. Therefore, if we compare with the performance of the SVM classifier itself, overall performance is worse. YOLOv3 was trained using the Common Objects in Context (COCO) dataset [49], which did not have enough lying-position persons for training the CNN to recognize persons in that position with high accuracy.

4.5. Experiment 3: Fall Detection with Pose Correction

YOLOv3 performance to detect persons in lying-positions improves with customized training using a dataset with a large number of persons in that position. However, to build this kind of dataset is costly and time-consuming. Due to the lack of public datasets with this characteristic at the moment, this training is not possible. The FPDS dataset proposed in this paper is useful for evaluating the robustness of the algorithm in different situations but does not have enough images for the customized training of YOLOv3.

The smallness of the training set represents a significant problem to the overall algorithm, as we analyzed in the previous experiment. Many have, therefore, tried to reduce the need for large training sets. In this article, we investigated how person pose position affects the efficiency of the approach. The experiments show that adding simple pose correction to YOLOv3 improves performance without the need for new customized training. The pose correction algorithm is explained in Figure 14. We ran three separate YOLOv3 networks, one for the initial image and two more for the rotated images at 90 and 270 degrees.

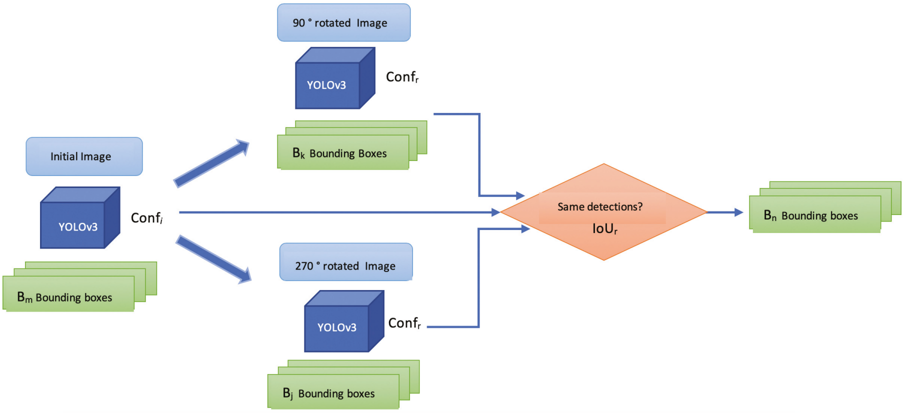


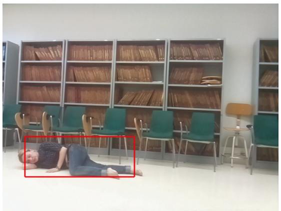
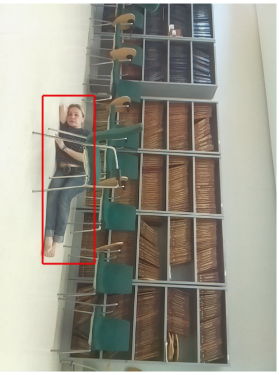
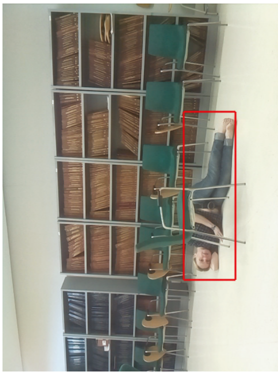
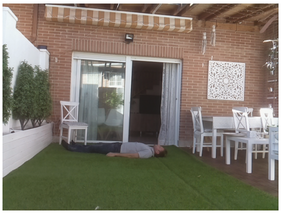
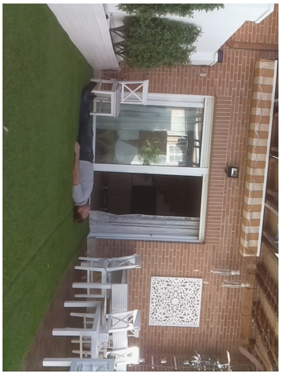
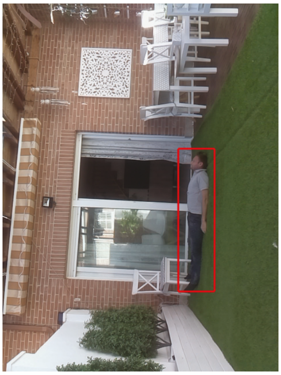
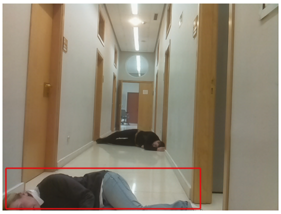
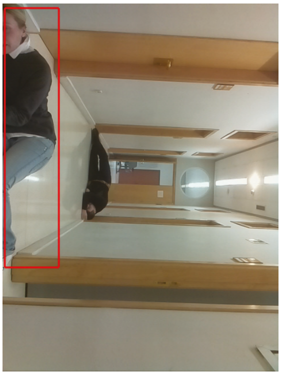
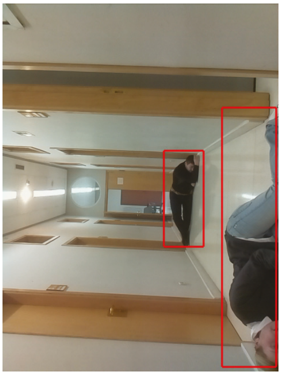
Figure 14. Flowchart of fall detection with pose correction.

For better optimization, we analyzed whether the correct threshold of the confidence score applied to the rotated images, called conf$_r$, was the same as the one used for the image directly from the camera, conf$_r$. Figure 13b shows the precision and recall metrics for different thresholds. In this case, we obtained the best trade-off for a value of conf$_r$ = 0.15, keeping the value of conf$_i$ = 0.2 for the image directly from the camera.

This modified person-detector algorithm could detect the same fall more than once. To identify if the bounding boxes belonged to the same fall, we needed to establish a new threshold for the IoU parameter, called IoU$_r$. In case the bounding boxes are the same, the algorithm keeps only one; otherwise, it keeps both of them. A threshold of IoU$_r$ = 0.1 provides a good trade-off between the precision and recall metrics.

Table 4 shows three examples from the testing set of the FPDS dataset with its detections in the initial and rotated images. In the first row, we can observe how the lying-person was detected in the two rotated images but not in the initial one. However, in the second example, the person was only detected in the 270°-rotated image. In the last example, with two fallen persons, one of the falls was detected in the three images but the other one was only detected in the 270°-rotated image.

Table 4. Fall detection examples with pose correction.

Initial Image	90° Rotated Image	270° Rotated Image
		
		
		

Thanks to pose position correction, the overall method improved considerably in recall while keeping almost the same precision. Results are shown in the third row of Table 3. It has detected 360 true positives, 31 false negatives and 17 false positives with values of precision and recall of 95.49% and 92.07%, respectively.

4.6. Evaluation 1: Relocation for Doubtful Cases

One of the main points of the proposed approach is the ability of the robot to relocate itself when fall detection is doubtful. The relocation algorithm moves the robot depending on the size and position of the detected bounding box. In all cases, the robot moves to center the possible fall detection with proper dimensions. Figure 15 shows three different cases where the robot needs to move to get a better picture of the person.

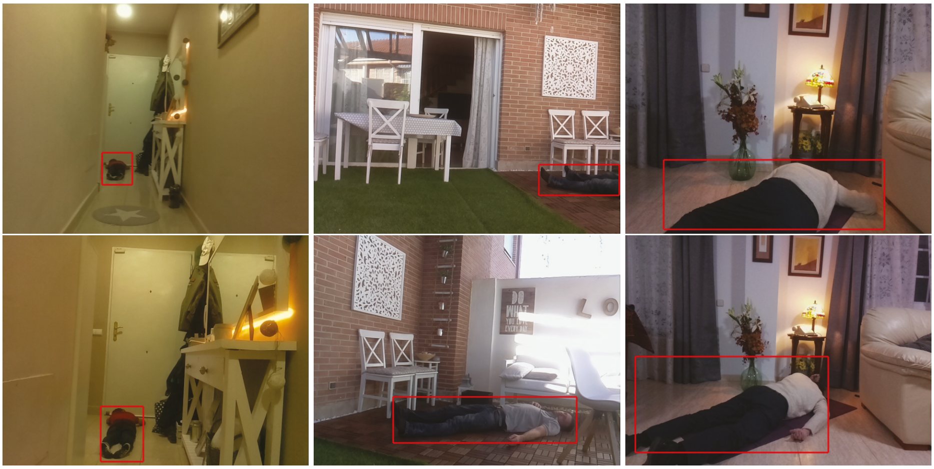


Figure 15. Robot relocation in three different testing examples of the FPDS dataset.

4.7. Evaluation 2: Other Datasets

To evaluate the detection effectiveness of our algorithm, we needed to test the proposed approach with alternative algorithms described in the literature. In this case, we decided to use the public Intelligent Autonomous Systems Laboratory Fallen Person Dataset (IASLAB-RGBD) [52], close enough to our dataset. This dataset was generated by using a Kinect One V2 camera mounted on a mobile robot 1.16 m above the floor. We used static dataset with 374 images, 363 falls and 133 nonfalls. Despite our camera being 76 cm above the floor and the training set having been built by using the same splits of the FPDS dataset as in the other test experiments, results were quite satisfactory in Experiment 1, with precision and recall of 99.45% and 100%, respectively. However, detection was not so good in Experiments 2 and 3, as can be observed in Table 5. These results indicate the good selection of the input feature vector to the SVM, which makes the classifier almost independent of the camera setup. Giving the impossibility to compare the results directly, the comparison is proof of the good performance of our method with other datasets that contain images that considerably differ from the examples in the training set.

Table 5. Performance over the Intelligent Autonomous Systems Laboratory Fallen Person Dataset (IASLAB-RGBD).

	TP	*FN*	*FP*	*Pr* (%)	*Re* (%)
Experiment 1: Fall classification	363	0	2	99.45	100
Experiment 2: Fall detection algorithm	212	151	43	83.13	58.40
Experiment 3: Fall detection with pose correction	271	92	53	83.69	74.72

5. Conclusions and Future Work

In this paper, we presented a low-cost system for detecting falls in elderly populations and people with functional disabilities who are living alone. The system is based on an assistive patrol robot that can be used for one, two, or more people. Our objective was to implement a vision-based fall detector system in our robot that acquires image data, detects falls and alerts emergency services. In our attempts to detect falls with an easy, fast and flexible end-to-end solution, we proposed a two-step algorithm. We combined a CNN to be used for person-detection and an SVM for fall classification. One of the main contributions of this paper was to find the combination features for the SVM-based classifier that provide the best performance for the design requirements. Results obtained from the different experiments indicate that the system had a high success rate in fall detection and could correct the position of the robot in case of doubt.

It is important to remark that, compared with existing fall detection approaches that show weakness in distinguishing between a resting position and a real fall scene, our fall classification algorithm could correctly detect both situations in all tested cases. Another important result to highlight is the ability to work correctly and detect fall situations with persons of different heights.

Since one of the goals of the work was to run a fall detection algorithm in real-time, it was needed to evaluate time implementation. In our case, the only time-consuming task was due to YOLOv3 person detection, which is more than acceptable for a real-time fall detection system.

We evaluated the robustness of the method using a realistic dataset called FPDS, which is publicly available and a contribution of this paper. The main features of this dataset are eight different scenarios, various person sizes, more than one person in an image, several lying-position perspectives and resting persons.

Additionally, we tested our algorithm using other datasets (training was done using the FPDS dataset). The results are quite satisfactory in fall classification, which showed us the almost-independence of the algorithm with the camera setup.

Future works to investigate are improvement in occlusion detection and the possibility to merge person detection and fall classification into a single CNN by using one or two different classes.

Author Contributions: Conceptualization, S.M.-B.; methodology, S.M.-B. S.L.-A., and C.I.-I.; software, C.I.-I.; validation, C.I.-I., S.M.-B., and P.M.-M.; writing—original-draft preparation, P.M.-M.; supervision, S.M.-B.

Funding: This work was partially supported by the Alcalá University research program through projects CCG2018/EXP-061 and TEC2016-80326-R from the "Ministerio de Economía, Industria y Competitividad".

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neuronal Network
RCCN	Region-based Convolutional Neuronal Network
YOLO	You Only Look Once
SVM	Support Vector Machine
RBF	Radial Basis Function
FPDS	Fallen Person DataSet
TP	True Positive
FN	False Negative
FP	False Positive

References

1. Ambrose, A.F.; Paul, G.; Hausdorff, J.M. Risk factors for falls among older adults: A review of the literature. *Maturitas* **2013**, *75*, 51–61. [CrossRef]
2. Rubenstein, L.Z. Falls in older people: Epidemiology, risk factors and strategies for prevention. *Age Ageing* **2006**, *35*, ii37–ii41. [CrossRef] [PubMed]
3. World Health Organization. WHO Global Report on Falls Prevention in Older Age. 2007. Available online: https://www.who.int/violence_injury_prevention/publications/other_injury/falls_prevention.pdf?ua=1 (accessed on 18 August 2019)

4. Noury, N.; Fleury, A.; Rumeau, P.; Bourke, A.; Laighin, G.; Rialle, V.; Lundy, J. Fall detection-principles and methods. In Proceedings of the 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 22–26 August 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 1663–1666.
5. Mubashir, M.; Shao, L.; Seed, L. A survey on fall detection: Principles and approaches. *Neurocomputing* **2013**, *100*, 144–152. [CrossRef]
6. Igual, R.; Medrano, C.; Plaza, I. Challenges, issues and trends in fall detection systems. *Biomed. Eng. Online* **2013**, *12*, 66. [CrossRef]
7. Khan, S.S.; Hoey, J. Review of fall detection techniques: A data availability perspective. *Med. Eng. Phys.* **2017**, *39*, 12–22. [CrossRef]
8. Tamura, T.; Yoshimura, T.; Sekine, M.; Uchida, M.; Tanaka, O. A wearable airbag to prevent fall injuries. *IEEE Trans. Inf. Technol. Biomed.* **2009**, *13*, 910–914. [CrossRef]
9. Bianchi, F.; Redmond, S.J.; Narayanan, M.R.; Cerutti, S.; Lovell, N.H. Barometric pressure and triaxial accelerometry-based falls event detection. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2010**, *18*, 619–627. [CrossRef]
10. Tmaura, T.; Zakaria, N.A.; Kuwae, Y.; Sekine, M.; Minato, K.; Yoshida, M. Quantitative analysis of the fall-risk assessment test with wearable inertia sensors. In Proceedings of the 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka, Japan, 3–7 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 7217–7220.
11. Rucco, R.; Sorriso, A.; Liparoti, M.; Ferraioli, G.; Sorrentino, P.; Ambrosanio, M.; Baselice, F. Type and location of wearable sensors for monitoring falls during static and dynamic tasks in healthy elderly: A review. *Sensors* **2018**, *18*, 1613. [CrossRef] [PubMed]
12. Mastorakis, G.; Makris, D. Fall detection system using Kinect's infrared sensor. *J. Real-Time Image Process.* **2014**, *9*, 635–646. [CrossRef]
13. Stone, E.E.; Skubic, M. Fall detection in homes of older adults using the Microsoft Kinect. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 290–301. [CrossRef]
14. Sumiya, T.; Matsubara, Y.; Nakano, M.; Sugaya, M. A mobile robot for fall detection for elderly-care. *Procedia Comput. Sci.* **2015**, *60*, 870–880. [CrossRef]
15. Zigel, Y.; Litvak, D.; Gannot, I. A method for automatic fall detection of elderly people using floor vibrations and sound—Proof of concept on human mimicking doll falls. *IEEE Trans. Biomed. Eng.* **2009**, *56*, 2858–2867. [CrossRef]
16. Martinelli, A.; Tomatis, N.; Siegwart, R. Simultaneous localization and odometry self calibration for mobile robot. *Auton. Robot.* **2007**, *22*, 75–85. [CrossRef]
17. Zhang, T.; Zhang, W.; Qi, L.; Zhang, L. Falling detection of lonely elderly people based on NAO humanoid robot. In Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA), Ningbo, China, 1–3 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 31–36.
18. Máthé, K.; Buşoniu, L. Vision and control for UAVs: A survey of general methods and of inexpensive platforms for infrastructure inspection. *Sensors* **2015**, *15*, 14887–14916. [CrossRef]
19. Iuga, C.; Drăgan, P.; Bușoniu, L. Fall monitoring and detection for at-risk persons using a UAV. *IFAC-PapersOnLine* **2018**, *51*, 199–204. [CrossRef]
20. Zhang, Z.; Conly, C.; Athitsos, V. A survey on vision-based fall detection. In Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments, Corfu, Greece, 1–3 July 2015; ACM: New York, NY, USA, 2015; p. 46.
21. Debard, G.; Mertens, M.; Deschodt, M.; Vlaeyen, E.; Devriendt, E.; Dejaeger, E.; Milisen, K.; Tournoy, J.; Croonenborghs, T.; Goedemé, T.; et al. Camera-based fall detection using real-world versus simulated data: How far are we from the solution? *J. Ambient. Intell. Smart Environ.* **2016**, *8*, 149–168. [CrossRef]
22. Kim, K.; Chalidabhongse, T.H.; Harwood, D.; Davis, L. Real-time foreground–background segmentation using codebook model. *Real-Time Imaging* **2005**, *11*, 172–185. [CrossRef]
23. Charfi, I.; Miteran, J.; Dubois, J.; Atri, M.; Tourki, R. Optimized spatio-temporal descriptors for real-time fall detection: comparison of support vector machine and Adaboost-based classification. *J. Electron. Imaging* **2013**, *22*, 041106. [CrossRef]
24. Liu, C.L.; Lee, C.H.; Lin, P.M. A fall detection system using k-nearest neighbor classifier. *Expert Syst. Appl.* **2010**, *37*, 7174–7181. [CrossRef]

25. Wang, S.; Zabir, S.; Leibe, B. Lying pose recognition for elderly fall detection. *Robot. Sci. Syst. VII* **2012**, *29*, 345.
26. Nait-Charif, H.; McKenna, S.J. Activity summarisation and fall detection in a supportive home environment. In Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, 26 August 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 4, pp. 323–326.
27. Rougier, C.; Meunier, J.; St-Arnaud, A.; Rousseau, J. Robust video surveillance for fall detection based on human shape deformation. *IEEE Trans. Circuits Syst. Video Technol.* **2011**, *21*, 611–622. [CrossRef]
28. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
29. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
30. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
31. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
32. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 379–387.
33. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
34. Li, Z.; Peng, C.; Yu, G.; Zhang, X.; Deng, Y.; Sun, J. Light-head r-cnn: In defense of two-stage object detector. *arXiv* **2017**, arXiv:1711.07264.
35. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
36. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
37. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
38. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
39. Laptev, I.; Caputo, B. Recognizing human actions: A local SVM approach. In *Null*; IEEE: Piscataway, NJ, USA, 2004; pp. 32–36.
40. Zhang, H.; Berg, A.C.; Maire, M.; Malik, J. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; IEEE: Piscataway, NJ, USA, 2006; Volume 2, pp. 2126–2136.
41. Ebrahimi, M.; Khoshtaghaza, M.; Minaei, S.; Jamshidi, B. Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* **2017**, *137*, 52–58. [CrossRef]
42. Burges, C.J. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [CrossRef]
43. Belousov, A.; Verzakov, S.; Von Frese, J. A flexible classification approach with optimal generalisation performance: Support vector machines. *Chemom. Intell. Lab. Syst.* **2002**, *64*, 15–25. [CrossRef]
44. Rougier, C.; Meunier, J.; St-Arnaud, A.; Rousseau, J. Fall detection from human shape and motion history using video surveillance. In Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07), Niagara Falls, ON, Canada, 21–23 May 2007; IEEE: Piscataway, NJ, USA, 2007; Volume 2, pp. 875–880.
45. Willems, J.; Debard, G.; Vanrumste, B.; Goedemé, T. A video-based algorithm for elderly fall detection. In Proceedings of the World Congress on Medical Physics and Biomedical Engineering, Munich, Germany, 7–12 September 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 312–315.

46. Svoboda, T.; Pajdla, T.; Hlaváč, V. Epipolar geometry for panoramic cameras. In Proceedings of the European Conference on Computer Vision, Germany, 2–6 June 1998; Springer: Berlin/Heidelberg, Germany, 1998; pp. 218–231.
47. Igual, R.; Medrano, C.; Plaza, I. A comparison of public datasets for acceleration-based fall detection. *Med. Eng. Phys.* **2015**, *37*, 870–878. [CrossRef] [PubMed]
48. Martínez-Villaseñor, L.; Ponce, H.; Brieva, J.; Moya-Albor, E.; Núñez-Martínez, J.; Peñafort-Asturiano, C. UP-fall detection dataset: A multimodal approach. *Sensors* **2019**, *19*, 1988. [CrossRef]
49. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 740–755.
50. Auvinet, E.; Rougier, C.; Meunier, J.; St-Arnaud, A.; Rousseau, J. *Multiple Cameras Fall Dataset*; DIRO-Université de Montréal, Tech. Rep; Université de Montréal: Montreal, QC, Canada, 2010; Volume 1350.
51. Kwolek, B.; Kepski, M. Human fall detection on embedded platform using depth maps and wireless accelerometer. *Comput. Methods Programs Biomed.* **2014**, *117*, 489–501. [CrossRef]
52. Antonello, M.; Carraro, M.; Pierobon, M.; Menegatti, E. Fast and robust detection of fallen people from a mobile robot. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 4159–4166.

electronics

MDPI

Article

Online Learned Siamese Network with Auto-Encoding Constraints for Robust Multi-Object Tracking

Peixin Liu, Xiaofeng Li *, Han Liu and Zhizhong Fu

School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; pxl@std.uestc.edu.cn (P.L.); hanliu@std.uestc.edu.cn (H.L.); fuzz@uestc.edu.cn (Z.F.)

* Correspondence: xfli@uestc.edu.cn; Tel.: +86-028-61830690

Received: 3 May 2019; Accepted: 22 May 2019; Published: 28 May 2019

Abstract: Multi-object tracking aims to estimate the complete trajectories of objects in a scene. Distinguishing among objects efficiently and correctly in complex environments is a challenging problem. In this paper, a Siamese network with an auto-encoding constraint is proposed to extract discriminative features from detection responses in a tracking-by-detection framework. Different from recent deep learning methods, the simple two layers stacked auto-encoder structure enables the Siamese network to operate efficiently only with small-scale online sample data. The auto-encoding constraint reduces the possibility of overfitting during small-scale sample training. Then, the proposed Siamese network is improved to extract the previous-appearance-next vector from tracklet for better association. The new feature integrates the appearance, previous, and next stage motions of an element in a tracklet. With the new features, an online incremental learned tracking framework is established. It contains reliable tracklet generation, data association to generate complete object trajectories, and tracklet growth to deal with missing detections and to enhance the new feature for tracklet. Benefiting from discriminative features, the final trajectories of objects can be achieved by an efficient iterative greedy algorithm. Feature experiments show that the proposed Siamese network has advantages in terms of both discrimination and correctness. The system experiments show the improved tracking performance of the proposed method.

Keywords: multi-object tracking; Siamese network; discriminative feature; online learning

1. Introduction

As a key technology in computer vision, multi-object tracking (MOT) has received growing attentions from researchers all over the world. In recent years, with the improvements in object detecting techniques [1–3], tracking-by-detection (TBD) has become one of the most successful strategies. It applies an object detector to produce detection responses in each frame, which are then used to generate complete trajectories. The data association process mainly depends on object features including appearance, motion, and other factors. It is often solved by Hungarian algorithms [4,5], network flows [6–8], minimum energy models [9,10], conditional random field approaches [11,12], hyper-graph model [13], deep learning methods [14–17], and so on.

Object feature expression is the basis of data association. Handcrafted features, such as the histogram of oriented gradient (HOG) [18], local binary patterns (LBP) [19], and the histogram of color (HOC) are widely used in computer vision researches [8,11,13,20]. These features were originally designed to distinguish objects from various backgrounds. Although a combination of different handcrafted features [11,13] is often used to improve discrimination, it is still not robust enough. Meanwhile, detection responses given by object detectors are not always accurate and sometimes even false due to complex backgrounds, poor image quality, complicated movements, or occlusions of objects. Thus, how to better distinguish targets by online detection responses, how to deal with noise due to detection inaccuracy, and how to combine various cues of a target to enhance discrimination remain key issues that limit tracking performance.

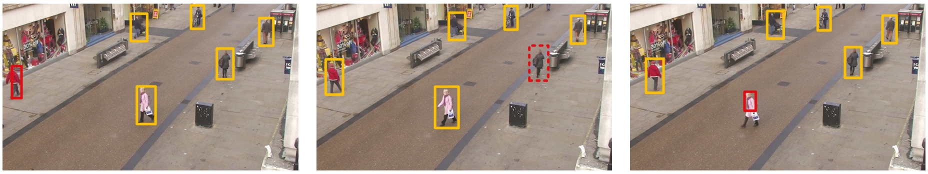
With the developments of deep learning in image classification, segmentation, and other applications, researchers used deep architectures to learn discriminative features for multi-object tracking, and they achieved good results. In [12,15,17,21–23], deep Siamese networks were adopted instead of traditional handcrafted methods [11,13]. A contrastive loss function was used with the aim of decreasing the feature distances for the same object pairs while increasing distances for the different pairs. Due to the shortage of online samples, training of such deep neural network mainly depends on offline learning. Although online fine-tuning measures are often adopted, the online data are too limited to run a deep network effectively.

In this paper, a Siamese network with an auto-encoding constraint (SNAC) is proposed, which is able to work well with a small-sized sample set. Different from previous deep Siamese networks, the SNAC has a simple structure with two fully-connected layers, an auto-encoder layer, and a code-mix layer. The simple network can be easily learned by online limited samples to extract discriminative features to distinguish objects on the scene. Inspired by stacked auto-encoder methods [24,25], the output of the encoder layer tries to represent the input detection response as accurately as possible. This is done by adding a constraint term to the loss function, called the auto-encoding constraint, which effectively prevents the network from overfitting while training with limited samples. To deal with inaccurate detection responses (red bounding box in Figure 1a), Gaussian distribution training samples are generated around detection responses to suppress noises. For each detection response, one SNAC is trained to distinguish it from others in adjacent frames. Meanwhile, in order to enhance robustness, following [22], the HOC is used as the input instead of raw pixels. With the discriminative detection features extracted by SNACs, reliable tracklets are generated.

To better distinguish tracklets, SNAC is improved to extract a composite previous-appearance-next (PAN) feature for each tracklet, which combines previous and next step motions with the appearance of the tracklet element. Following [11,26], elements in the same tracklet can be treated as positive samples, and the negative samples are obtained from time overlapped tracklets. The distribution is proposed to express motion that can suppress motion noises, and this is also compatible with the appearance for joint learning of the PAN feature.

In order to solve the MOT problem by the proposed SNAC, an online incremental learned tracking framework is established. First, one SNAC is trained for each detection response online, and reliable tracklets are generated mainly by the extracted features. Then, the PAN features are learned from tracklets by improved SNACs. To improve the training efficiency, SNACs are trained by incremental learning. During tracklet generation, the parameters of SNAC for detection in the new frame are inherited from the predecessor tracklet element, and the training samples are updated frame by frame. To extract PAN, the parameters are initialized by the SNAC of the related detection response. A tracklet growing process is used to deal with missing and partial detections (Figure 1b,c) before tracklet association. With the discriminative PAN feature, complete trajectories are solved efficiently by an iterative greedy algorithm. The main contributions of this paper are summarized as follows:

(1) A simple structure Siamese network with an auto-encoding constraint is proposed to extract discriminative features efficiently for objects on the scene. An auto-encoding constraint is added to prevent overfitting when training data are limited.
(2) A composite feature of tracklet, PAN, is defined, which combines appearance and motion through joint learning. To describe the sequential features of tracklets better, data association based on PAN is more reliable.
(3) A tracking framework is established that includes reliable tracklet generation by incremental learning with SNAC for the detection response, tracklet growth to enhance PAN performance and deal with missing detections, and tracklet association with PAN to generate complete trajectories.



(**a**) Inaccurate detection (**b**) Missing detection (**c**) Partial detection

Figure 1. Illustrations of detection failures in three consecutive frames. The solid yellow bounding boxes represent the correct detection responses, and the red boxes are error cases. (**a**) The red bounding box is a deviation detection that does not exactly match the target. (**b**) The red dashed bounding box indicates a missing detection. (**c**) The detection response only includes the upper body of the target.

2. Related Works

Tracking by detection (TBD) has been one of the most promising methods developed to solve the multi-object tracking (MOT) problem in recent years. It generates object trajectories based on detection responses given by pre-designed detectors. For reliable data association, most recent researches were based on tracklets. In [26], the dual-threshold method was proposed to generate reliable tracklets and utilize them to get the final trajectories hierarchically. In [27], a prototype of a three frames triplet, which is a type of three members tracklet, was designed to extract high-level features. The Hungarian algorithm was also used to generate reliable tracklets in [12,15]. On the basis of tracklets, [11] built an online learning conditional random field (CRF) model focused on distinguishing the difficult pairs of objects. In [13], a hyper-graph model was developed to explore more complex relations among objects. The latest MOT methods [12,21] also focused on using tracklets. In these studies, tracklet building and feature expression are important to achieve reliable data association. In this section, MOT object feature extraction methods are mainly introduced.

From handcrafted methods to deep learning techniques, many studies have achieved significant improvements in extracting appropriate object features for MOT. In [11,13], a combination of multiple handcrafted features was proposed to distinguish objects by appearance. Their sample collection schemes were used in many following studies. The developments of deep learning have introduced new ideas for feature description in tracking areas. In [24,28–30], deep neural networks were adopted for single object tracking (SOT), and achieved significant improvements. In SOT problems, features of objects were used to distinguish them from the background. Different from SOT, MOT mainly distinguishes objects from each other. Due to this difference, the deep learning scheme of SOT cannot provide good results for MOT problems.

The deep learning methods for MOT can be summarized into two categories. The first builds a deep learning based tracking model to form the whole MOT system. Milan et al. [31] proposed a tracking model based on recurrent neural networks (RNN). The proposed RNN model described the whole tracking system including motion prediction, updating, object state judgment, and data association. It was trained online in an end-to-end manner to track various objects. Schulter et al. [14] proposed a deep network flow model for MOT, which instead of empirically hand-crafting costs, learned the parameterized costs of the network flow model by end-to-end training. This dynamic parameter setting method improved the robustness and accuracy of tracking. Zhou et al. [12] proposed a deep continuous conditional random field (DCCRF) model for solving online MOT problems. The unary term was used to provide a deep discriminative appearance feature for tracklet association, and a pairwise term was used to deal with inter-object relations. In [16], a deep neural network consisting of an encoder and a decoder was proposed. In their method, an encoder was a fully-connected network and a decoder was a bidirectional long short-term memory (LSTM). This network was able to learn the association matrix to solve MOT.

The second group uses a deep neural network to extract discriminative feature for each object. Unlike the previous kind, this method deals with the object feature extraction problem directly, and many researchers have followed this idea. Sadehgian et al. [32] proposed an RNN model jointly used the appearance, motion, and interactions of an object to encode a discriminative long-term temporal relationship using these cues. Their discriminative appearance features were extracted by a deep CNN. Son et al. [33] designed a quadruplet CNN (QCNN) network to learn the affinities among objects based on appearance and motion. The proposed quadruplet loss function guided the network to learn a temporally-smooth appearance model with motion-aware constraints. Features extracted from the QCNN included time continuity, which enhanced the discrimination. In addition, Siamese networks, first defined and used for signature verification, played an important role and have achieved good results in face identification [34], people re-identification [35], and many computer vision applications. Siamese networks are more suitable for distinguishing objects due to their symmetrical structures. Wang et al. [15] applied a Siamese CNN (SCNN) to construct an appearance affinity model for tracklets. They embedded a temporally-constrained multi-task mechanism in their training process. Leal-Taixé et al. [22] used an SCNN to estimate the likelihood of two objects using a multi-modal inputs including image and optical flow. Following [22], Yoon et al. [23] proposed the historical appearance matching method and trained a Siamese network by a two-step process to deal with noisy detections. In [17], a speeding method was proposed to remove redundant appearance matchings of SCNN for real-time tracking. In the DCCRF model [12], SCNN was also used to extract discriminative features. Based on SCNN, Bae et al. [21] proposed a confidence-based data association method for MOT. They utilized the SCNN to learn a discriminative appearance model from offline training datasets.

3. Online Learned Siamese Network with Auto-Encoding Constraint

In this section, a new Siamese network with an auto-encoding constraint (SNAC) is proposed. It is better at distinguishing objects in MOT. Benefiting from the simple structure of two fully-connected layers, an auto-encoder layer and a code-mix layer, the SNAC can be learned effectively. Meanwhile, with an auto-encoding constraint in the loss function, SNAC can prevent overfitting while training with limited online samples. In order to suppress detection noises, Gaussian distribution samples were generated around detection responses to make up the training set and HOC was used as the input instead of raw pixels. Then, an incremental learning algorithm was proposed to train the SNAC to generate reliable tracklets. Mathematical notations are listed in Table 1.

Table 1. Notations.

Symbol	Definition
SNAC	Siamese network with an auto-encoding constraint
d_i^t	the i^{th} detection response in frame t
D_t	detection responses set in frame t
$\mathbb{D} = \{D_1, ..., D_t\}$	sequence of D_i for $i = 1, 2, \ldots, t$
T_k^t	the k^{th} tracklet up to frame t
$\mathbb{T}^t = \{T^1, ..., T^t\}$	set of all tracklets up to frame t
$\mathbf{F}_t^k$	feature vector of SNAC for T_k^t
$\{d\}$	a set consisting of an element d
$D_t - \{d\}$	the set D_t with d deleted
$\Psi(.)$	the sample set
$\Lambda_a(T, d)$	appearance similarity between T and d
$\Lambda(T, d)$	overall affinity between T and d

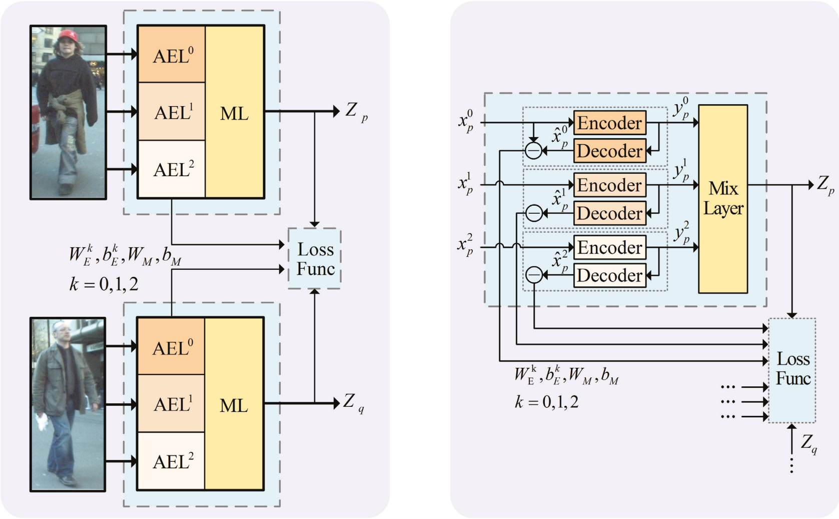
3.1. The Structure of SNAC

The two-layer structure of SNAC is shown in Figure 2a. Bounding boxes of detection responses were first resized to 48 × 32 as the inputs of the Siamese network. The two sub-networks (dashed boxes in Figure 2a) were identical in structure and share parameters including weights and biases. A contrastive loss function was employed to learn the Siamese network.

As shown in Figure 2b, each sub-network consisted of an auto-encoder layer and a code-mix layer. The first layer contained three parallel auto-encoders corresponding to the red, green, and blue channels of the input RGB image, respectively. Similar to [22], the inputs were R, G, and B histograms, not pixel values, and they were denoted as 256 dimensions vectors: **x**0, **x**1, and **x**2. Because of limited samples, training based on pixel values may lead to overfitting. Meanwhile, the histogram can also suppress the detection noises. Each auto-encoder contained a forward encoder, a backward decoder, and an auto-encoding error evaluator. The encoder and decoder were fully-connected networks. The output of the encoder was a vector with 100 dimensions, and the output of the decoder was a reproduction of the corresponding input. The code-mix layer was fully connecting and combined three code vectors of the first layer to produce a feature vector with 100 dimensions as the final output. Mathematically, the sub-network can be written as:

$$\begin{cases} \mathbf{y}_m^k = \sigma(\mathbf{W}_E^k \mathbf{x}_m^k + \mathbf{b}_E^k), m = p, q, k = 0, 1, 2 \\ \hat{\mathbf{x}}_m^k = \sigma(\mathbf{W}_D^k \mathbf{y}_m^k + \mathbf{b}_D^k), m = p, q, k = 0, 1, 2 \\ \mathbf{z}_m = \sigma(\mathbf{W}_M(\mathbf{y}_m^0, \mathbf{y}_m^1, \mathbf{y}_m^2) + \mathbf{b}_M), m = p, q \end{cases} \qquad (1)$$

where subscript m indexes the upper p or lower q sub-network, the upper-script k indexes the channel, **y** is the code vector from an encoder, $\hat{\mathbf{x}}$ is the reproduction of **y** by the decoder, and **z** is the final feature vector. **W**, **b**, and σ are the weights, biases, and activation functions of the neural networks, with the subscripts E, D, and M indicating the encoder, decoder, and code-mix layer.



(**a**) The overall framework

(**b**) Internal details

Figure 2. Structure of SNAC: (**a**) shows the overall structure of SNAC, including its symmetrical structure and parameter sharing. Here, AEL stands for auto-encoder layer, superscripts 0, 1, and 2 indicate image channel numbers, and ML stands for the code-mix layer. (**b**) is the internal anatomical diagram of the SNAC structure, showing its auto-encoder layer and code-mix layer.

3.2. Loss Function and Auto-Encoding Constraint

To learn a Siamese network, a contrastive loss function was formulated based on similarity or difference measurements between input pair. The objective was to train the network to sufficiently reduce differences between pairs of the same inputs and to increase feature distances of different ones. The distance of input training pair is denoted as:

$$D(\mathbf{x}_p, \mathbf{x}_q) = ||\mathbf{x}_p - \mathbf{x}_q||_2^2 \tag{2}$$

where $\mathbf{x}_p$ and $\mathbf{x}_q$ are feature vectors from the two sub-networks in SNAC. Instead of using the Euclidean distance here, other measures, like Mahalanobis and Bhattacharyya distances, can be used.

Given a group of training samples, the loss function of SNAC to be minimized consists of three terms, L1, L2, and L3, as follows:

$$\begin{aligned} L &= \alpha L1 + \beta L2 + \gamma L3 \\ &= \alpha \sum_{p,q} \max(0, \delta - l_{pq}[1 - ||\mathbf{z}_p - \mathbf{z}_q||_2^2]) \\ &+ \beta \sum_{k=0,1,2} ||\mathbf{x}_j^k - \hat{\mathbf{x}}_j^k||_2^2 \\ &+ \gamma(\sum_{k=0,1,2} ||\mathbf{W}_k||_2^2 + ||\mathbf{b}_k||_2^2) \end{aligned} \tag{3}$$

where α, β, and γ are weight coefficients between zero and one. The first term, L1, is a margin-based loss of difference of sample pairs; δ is the decision margin, which satisfies $(0 \leq \delta \leq 1)$; l_{pq} is the sample indicator; $l_{pq} = 1$ denotes a positive pair; and $l_{pq} = 0$ denotes a negative pair. The L3 term is the regularization constraint.

However, deep neural networks contain a large number of parameters and require huge sample sets for training. For the case of using limited online samples, parameters of a deep model will often be overfitting after training, and the network will not work. This method often pays more attentions to some local details of training samples and does not balance the general features. Subsequently, inspired by the stacked auto-encoder in [24,25], the L2 term was added, an auto-encoding constraint (AC) to the loss function in Equation (3), to prevent overfitting, even when training with limited online samples.

3.3. Denoising through the Collection of Training Samples

$D_t = \{d_i^t, i = 1, 2, ...N_t\}$ is the detection set at frame t. Each detection response d_i^t was associated with the SNAC(d_i^t). Training samples were collected around d_i^t. The purpose of SNAC(d_i^t) is to distinguish d_i^t from other object detection responses in adjacent frames, not over a longer time period. The training samples of SNAC(d_i^t) were collected online. Inspired by [11], d_i^t is the only one positive sample, and the remaining detection responses at frame t constitute the negative sample set. Although SNAC(d_i^t) can be trained by small-sized samples, an unbalanced sample set with only one positive sample cannot drive it. To solve this problem, more samples are needed, which means additional detection responses of d_i^t.

There is a fundamental issue whereby detection responses are not always perfect, and their bounding boxes are often inaccurate, as explained before in Figure 1a. When a noisy detection is used as a training sample, it will impair the parameters of SNAC. However, detection noise is inevitable, so this error can be suppressed through more d_i^t with random noise. This noise processing is just enough to solve the positive sample shortage problem.

Detection noise was assumed to be modeled as additive noise as follows:

$$\mathbf{p}_n = \mathbf{p} + \mathbf{n}_p,\ \mathbf{s}_n = \mathbf{s} + \mathbf{n}_s \tag{4}$$

where $\mathbf{p} = (x, y)$ is the center position of the detection response, $\mathbf{s} = (w, h)$ is the size vector of width and height, and $\mathbf{n}_p$ and $\mathbf{n}_s$ are additive noises that refer to position and size, respectively. $\mathbf{n}_p$ and $\mathbf{n}_s$ are assumed to follow a Gaussian distribution, $G(0, \sigma_p)$ and $G(0, \sigma_s)$, where σ_p and σ_s are corresponding covariances obtained by prior analysis.

A group of random bounding boxes $\Psi(\{d_i^t\})$ was generated around d_i^t according to Equation (4) with distributions of $\mathbf{n}_p$ and $\mathbf{n}_s$. In the same way, $\Psi(D_t - \{d_i^t\})$ was obtained. $\Psi(\{d_i^t\})$ and $\Psi(D_t - \{d_i^t\})$ are the positive and negative sample sets, respectively. Using these online collected samples, $\text{SNAC}(d_i^t)$ not only can extract discriminative features for d_i^t, but it also can suppress detection noises.

3.4. Iterative Tracklet Generation with SNAC by Incremental Learning

The above sections discussed the establishment and training of SNAC. Each detection response d_i^t is associated with $\text{SNAC}(d_i^t)$, which extracts discriminative features to better distinguish d_i^t from other detections belonging to D_{t+1}. Moreover, connecting these original independent networks not only increases the number of samples, but can also improve the training efficiency. On the one hand, $\text{SNAC}(d_i^t)$ can obtain more training samples from d_j^{t-1} in the adjacent frame $t-1$ through a relationship. On the other hand, with this relationship, $\text{SNAC}(d_i^t)$ does not need random initialization parameters for training, but inherits them from $\text{SNAC}(d_j^{t-1})$, which can reduce the training time to improve the efficiency. This relationship is the principle of tracklet linking, that is the two detection responses between adjacent frames belong to the same object. Incremental learning of SNACs through this inheritance relationship can effectively match adjacent frame detection responses. To generate reliable tracklets, an iterative algorithm with SNAC by incremental learning is proposed as shown in Algorithm 1.

Algorithm 1 Iterative tracklet building with SNAC by incremental learning.

```
Input: $\mathbb{D} = \{D_1, D_2, ..., D_t\}$, detection set of each frame
Output: $\mathbb{T}^t = \{T_k^t\}$, tracklet setup to frame $t$
Initialization: $t = 1$, $\mathbb{T}^1 = \emptyset$
for each $d \in D_1$ do
    $T_k^1 = d$
    Initialize $\mathbf{F}_k^1$ with random parameters
    Set $P = \Psi(d)$, $N = \Psi(D_1 - d)$
    Train $\mathbf{F}_k^1$ with $P$ and $N$
end for
while $t \geq 2$ do
    for each $T_k^{t-1} \in \mathbb{T}^{t-1}$ and each $d \in D_t$ do
        Compute $\Lambda_a(T_k^{t-1}, d)$ as Equation (6)
        Compute $\Lambda(T_k^{t-1}, d)$ as Equation (5)
    end for
    For all $\Lambda(T_k^{t-1}, d)$ meeting the link requirement, select
    pairs of $T_k^{t-1}$ and $d$ by the Hungarian algorithm.
    $\mathbb{T}^t$ = renewed $\mathbb{T}^{t-1}$ by linking the selected pairs.
    $D_t^R = D_t$
    for each $T_k^t \in \mathbb{T}^t$ having a new detection added do
        $d$ = the new detection of $T_k^t$
        Set $P = \Psi(d)$, $N = \Psi(D_t - d)$
        $\mathbf{F}_k^t = \mathbf{F}_k^{t-1}$ incrementally trained with $P$ and $N$
        $D_t^R = D_t - d$
    end for
    for each $d \in D_t^R$ do
        Add a new single member tracklet $T_k^t = d$,
        and set its $\mathbf{F}_k^t$ as above.
    end for
end while
```

At the first frame $t = 1$, a new tracklet T_i^1 was established by a single member of d_i^1 in D_1, and the current total number of tracklets was N_1. To match the detection response belonging to the same object (or inexistence) in the next frame, a randomly initialized network, SNAC(d_i^1), was associated with d_i^1. After SNAC(d_i^1) training, the appearance similarity $\Lambda_a(T_i^1, d_j^2)$ can be calculated by T_i^1, which is equal to d_i^1 and d_j^2. Together with the position similarity $\Lambda_p(T_i^1, d_j^2)$ based on position and size, the total similarity $\Lambda(T_i^1, d_j^2)$ can be calculated. When similarities of all detection responses in Frame 1 have been calculated, the Hungarian algorithm was used to determine if there was a d_j^2 that could be combined with T_i^1. If d_i^1 and d_j^2 belong to the same object, d_j^2 joins with T_i^1, and tracklet T_i^1 is updated to T_i^2. Otherwise, a new tracklet $T_{N_1+1}^2$ of d_j^2 is generated. Then, the processing went into Frame 2, and tracklets that contained the detection responses in Frame 2 needed to train. Taking T_i^2 as an example, its last element was d_j^2. If d_i^1 exists as a former element of d_j^2 in tracklet T_i^2, the initial parameters of SNAC(T_i^2) equal to SNAC(d_j^2) will be inherited from the trained SNAC(d_i^1). In addition, the positive and negative training sets can be expanded through the samples of SNAC(d_i^1). Training of SNAC(T_i^2) can be done with fewer iterations in this incremental manner. If T_i^2 is a new added tracklet that only contains d_j^2, SNAC(T_i^2) will be trained similarly to SNAC(d_i^1). Finally, all reliable tracklets $\mathbb{T}$ will be produced frame-by-frame.

Now, the calculation of similarities between a tracklet and a detection response is explained. $\Lambda(T_k^{t-1}, d_j^t)$ is given as follows:

$$\Lambda(T_k^{t-1}, d_j^t) = \Lambda_a(T_k^{t-1}, d_j^t)\Lambda_o(T_k^{t-1}, d_j^t). \tag{5}$$

The appearance similarity was computed by the distance between feature vectors output by the SNAC(T_k^{t-1}). It is given by:

$$\Lambda_a(T_k^{t-1}, d_j^t) = g\{\|\mathbf{F}_k^{t-1}((T_k^{t-1}(e))) - \mathbf{F}_k^{t-1}(d_j^t)\|_2^2\} \tag{6}$$

where $T_k^{t-1}(e)$ denotes the end element of tracklet T_k^{t-1}, $\mathbf{F}_k^{t-1}$ denotes the output feature vector of the SNAC for tracklet T_k^{t-1}, and g is a probability function on the squared distance of feature vectors. Because of the margin-based loss of SNAC, the definition of function g is as follows:

$$g(x) = \begin{cases} 1 & x < 1 - \delta \\ 0 & x > 1 + \delta \\ (1 + \delta - x)/2\delta & otherwise \end{cases} \tag{7}$$

where δ is the decision margin given in the loss function of Equation (3).

Overlapping is widely used to describe the detection position relationship. It takes information about the coordinates and size into account. The overlapping $\Lambda_o(T_k^{t-1}, d_j^t)$ is given as:

$$\Lambda_o(T_k^{t-1}, d_j^t) = \frac{A_\cap(T_k^{t-1}(e), d_j^t)}{\min[A(T_k^{t-1}(e), A(d_j^t)]} \tag{8}$$

where A is the area function on a detection response and $A_\cap$ is the area function on the intersection of two detection responses.

4. Multi-Object Tracking Framework

4.1. Overall Framework

Based on SNAC, a tracking framework following TBD was established to solve the MOT problem. A TBD scheme can be described as solving an MAP problem by:

$$\mathcal{T}^{*} = \arg\max_{\mathcal{T}} P\left(\mathcal{T}|\mathbb{D}\right) \tag{9}$$

where $\mathbb{D}$ is the set of given detection responses and $\mathcal{T}$ is the set of trajectories. In the framework, tracklets were first generated. Because a tracklet is an ordered combination of detection responses, it is able to extract higher order features to better describe relations between objects. Then, the problem can be converted into a more reliable tracklet association as follows:

$$\mathcal{T}^{*} = \arg\max_{\mathcal{T}} P\left(\mathcal{T}|\mathbb{T}\right) \tag{10}$$

where $\mathbb{T}$ is the set of all tracklets.

The whole framework is shown in Figure 3. First of all, the inputs were checked, and deformity detection responses were deleted, such as too large or small bounding boxes. SNAC was proposed to extract discriminative appearance features for detection responses. The online SNAC incremental learning method mentioned above was used to generate reliable tracklets. The next step was to generate tracking results through tracklet association. Similar to detection association based on the learning method, SNAC was improved to extract a new discriminative composite feature PAN for the tracklet instead of using traditional handcrafted methods. To enhance tracklet association, the tracklet growing module was embedded to make tracklets as extended as possible. With the discriminative PAN feature, tracklet association was converted to a linear programming problem that was solved by an efficient greedy iterative algorithm, and the final trajectories were achieved. For real-time tracking, the whole tracking process was carried out in sliding time windows.

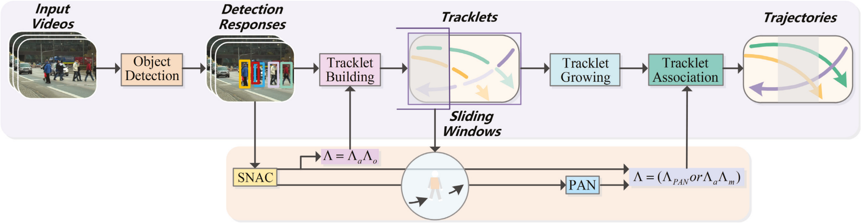


Figure 3. Illustration of the overall online tracking by detection (TBD) framework. In addition to standard inputs and outputs, an online tracking framework is established with new facilities, including an iterative Siamese network with an auto-encoding constraint (SNAC) to learn the detection responses, previous-appearance-next (PAN) to represent the composite features of tracklets, and pre-processing of tracklet growth to cope with short-time detection failures. Finally, a greedy iterative algorithm is used to output robust trajectories in sliding windows.

4.2. Previous-Appearance-Next Feature of the Tracklet

A tracklet $T_m^{t2} = \{d_i^{t1}, d_j^{t1+1}, ...d_k^{t2}\}$ is an ordered sequence of detection responses that represents a moving object with a short time from frame $t1$–$t2$. To describe T_m^{t2}, appearance and motion are indispensable. They are often assumed to be independent of each other in several studies [12,21,36]. Only by weighted summation can they express the similarity between two tracklets. To increase the flexibility and discrimination, a composite previous-appearance-next (PAN) feature was proposed.

The new feature combined appearance and motion for the tracklet, and it was extracted jointly by an improved SNAC.

Taking T_m^{t2} and T_n^{t4} as examples, as shown in Figure 4b, T_n^{t4} is from frame $t3$–$t4$ and $t2 < t3$. To calculate the similarity between T_m^{t2} and T_n^{t4}, it is better to use the tail part of T_m^{t2} and the head part of T_n^{t4} rather than using their whole information. $T_m^{t2}(e)$ is the last element of tracklet T_m^{t2}, and $T_n^{t4}(s)$ is the first element of T_n^{t4}. The PAN($T_m^{t2}(e)$) vector integrated the appearance, previous, and next stage motions of $T_m^{t2}(e)$ to express the tail part composite feature of tracklet T_m^{t2}. Correspondingly, the PAN($T_n^{t4}(s)$) vector was defined for the head part composite feature of T_n^{t4}. The next stage motion of tail T_m^{t2} and the previous of head T_n^{t4} were computed by estimation methods.

The SNAC for detection response was revised to extract PAN(.) vectors of tracklets. The new structure is shown in Figure 4a. The previous and next stage motions were used as additional inputs to the mix-layer. The first layer of the new SNAC was same as the old SNAC. $\mathbf{\Delta}^p = (x^p, y^p)$ and $\mathbf{\Delta}^n = (x^n, y^n)$ are the previous and next motion vectors of $T_m^{t2}(e)$, respectively. As shown in Figure 4b, $\mathbf{\Delta}^p$ represents the x and y axes displacements of T_m^{t2} from $t2 - 1$ to $t2$. For the next-stage motion vector, $T_m^{t2}(e+1)$, the estimation of T_m^{t2} in frame $t2 + 1$ was computed first, and then, $\mathbf{\Delta}^n$ of $T_m^{t2}(e)$ was calculated.

Since $\mathbf{\Delta}^p$ and $\mathbf{\Delta}^n$ are two-dimensional vectors that include displacements with x and y directions and the output of each auto-encoder in the first layer of SNAC is a 100-dimension feature vector, they are totally different in type and cannot work together simply. Meanwhile, the existence of detection noises makes the deterministic motion descriptions inaccurate. A distribution description method was proposed to represent the motion instead of specific values. Assuming following the Gaussian distribution, the x axis displacement, x^p of $\mathbf{\Delta}^p$ for instance, is described by $G(x^p, \sigma_x)$, where σ_x is set by pre-training. $G(y^p, \sigma_y)$ is for y displacement, as well. The distribution description was given by sample vectors of $G(x^p, \sigma_x)$ and $G(y^p, \sigma_y)$, and its length was taken to be equal to that of the appearance vector. For in MOT, the motion feature is as important as appearance. The distribution description for $\mathbf{\Delta}^n$ can also be obtained. Then, they were merged with the three outputs of the first layer to form one mixed vector for the second-layer training.

Then, SNAC($T_m^{t2}(e)$) was trained to extract the tail PAN($T_m^{t2}(e)$) feature. Training samples of SNAC($T_m^{t2}(e)$) were also collected online. Similar to [11], elements in T_m^{t2} are positive samples. Tracklets that overlap with T_m^{t2} in time are positive samples. The parameters of the first layer were inherited from the corresponding detection SNAC. After training the SNAC(T_m^{t2}), discriminative local composite features can be extracted to distinguish T_m^{t2} from other subsequent tracklets.

As shown in Figure 4b, similarities between tracklet T_m^{t2} and T_n^{t4} were computed. After training, PAN($T_m^{t2}(e)$) and PAN($T_n^{t4}(s+1)$), as shown by the blue dashed circle areas in the figure, were extracted. Then, forward similarity was achieved as follows:

$$S_{m,n}^F = ||PAN(T_m^{t2}(e)) - PAN(T_n^{t4}(s+1))||_2^2 \tag{11}$$

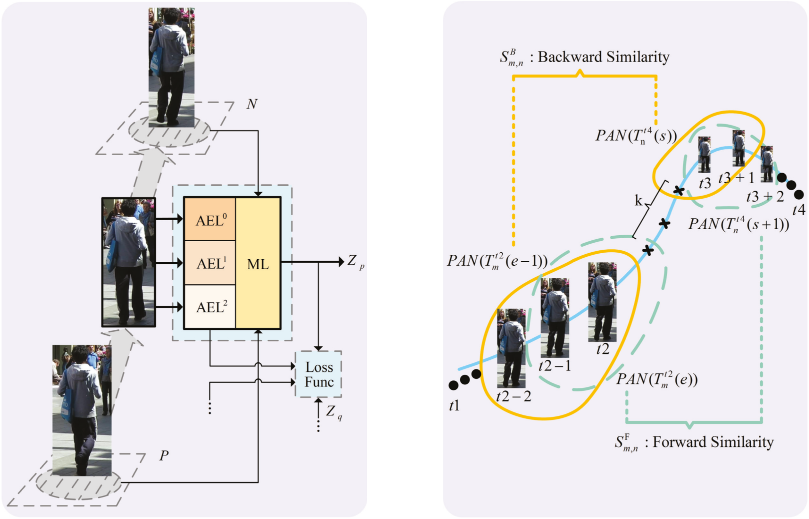
To get a reliable similarity, the backward relationship was also computed, as shown in Equation (12).

$$S_{m,n}^B = ||PAN(T_m^{t2}(e-1)) - PAN(T_n^{t4}(s))||_2^2 \tag{12}$$

The final similarity was given by:

$$\Lambda_{PAN}(T_m, T_n) = g(\min(S_{m,n}^F, S_{m,n}^B)) \tag{13}$$

where g is the probability function for the distance of feature vectors, as defined in Equation (7).



(**a**) SNAC for PAN

(**b**) Similarity computing by PAN

Figure 4. The generation and application of PAN. (**a**) SNAC is revised and added two pieces of motion information of a tracklet member together with the appearance codes from the auto-encoder layer as the inputs of the code-mix layer. During the online training process, the PAN feature is the final output of the code-mix layer. (**b**) Similarities of tracklets are determined by calculating the forward and backward PAN affinities.

4.3. Tracklet Growing

If the frame gap between T_m^{t2} and T_n^{t4} was small, variations in the appearance and motion from T_m^{t2}–T_n^{t4} were not obvious, and the PAN could work well. Otherwise, the long-term frame gap brought a large variety of appearances, and motions may reduce the performance. PAN considers more local elements of the tracklet to enhance the performance. In order to make tracklet association more reliable, it is effective to reduce the time interval in the sliding windows as much as possible. Therefore, the tracklet growing process was used to extend the tracklet by estimated bounding boxes, which were missing from the detection. It contained forward and backward growth.

To forward the extended tracklet T_m^{t2}, the center position $\mathbf{p}_1^f(T_m^{t2}) = (\hat{x}, \hat{y})$ in frame $t2+1$ was first estimated by quadratic fitting. Then, the optimal estimation bounding box was searched as follows:

$$\begin{aligned} d^{*} &= \arg\min_{d \in C} \left\| \mathbf{H}(T_m^{t2}(e)) - \mathbf{H}(d) \right\|_2^2 \\ s.t. & \left\| \mathbf{H}(T_m^{t2}(e)) - \mathbf{H}(d) \right\| \leq \varepsilon_1 \end{aligned} \tag{14}$$

where C is the candidate bounding boxes set, center positions x and y are sampled according to the distribution of $G(0, \sigma_m)$, and the size is equal to $T_m^{t2}(e)$. $\mathbf{H}$ denotes the color histogram of detection $T_m^{t2}(e)$. The goal was to find the most similar estimation. If the optimal estimation d_o^{t2+1} was found, a conflict process was also required to avoid false alarms. If the overlap between d_o^{t2+1} and an existing d_i^{t2+1} exceeded the threshold, the forward growth of T_m^{t2} stopped. Otherwise, T_m^{t2} was updated to T_m^{t2+1} with d_o^{t2+1} and the growing process continued to frame $t2+2$. The backward extension was similar to the forward process. For the isolated tracklets, random sampling was used to form the candidate estimations. After these missing detection compensation processes, tracklets were extended to improve the discrimination performance of PAN, and more reliable associations could be made.

4.4. Tracklet Association in Sliding Windows

Tracklet association was the last module in MOT to generate the final trajectories of objects. The main task was to link tracklets belonging to the same objects into a complete trajectory based on similarities among tracklets. Solutions such as min-cost networks, energy minimization, successive shortest paths, and the Hungary algorithm are widely used to generate tracking results. Global optimization is an ideal scheme because the previous judgments will be revised to achieve the overall optimal results. In cases where it is difficult to distinguish objects, this dynamic scheme can achieve better tracking performance than a greedy strategy. Similar to tracking by learning feature extraction method [15], network flows methods were no longer used to get the tracking result. The MAP problem shown in Equation (10) was directly mapped to a generalized linear assignment:

$$\begin{aligned} & \underset{L}{max} \sum_{i=1}^{N} \sum_{j=1}^{N} \Lambda(T_i, T_j) L_{ij} \\ & s.t. \sum_{i=1}^{N} L_{ij} \leq 1; \sum_{j=1}^{N} L_{ij} \leq 1 \end{aligned} \tag{15}$$

To solve problem Equation (15), the similarity $\Lambda(T_i, T_j)$ between tracklets was used; this is equal to linking probabilities mainly based on PAN features. $\Lambda(T_i, T_j)$ was computed by Equation (13). However, PAN features cannot be extracted from tracklets with lengths of less than two elements. For this particular case, $\Lambda(T_i, T_j)$ degenerated into the traditional weighted combination of appearance and motion. L_{ij} is the association indicator, where 1 indicates connection and 0 means disconnection. The constraints guaranteed the uniqueness of association. As the better discriminative PAN, the similarity matrix $\mathbf{\Lambda}$ was normalized, and Equation (15) was solved by a greedy iterative algorithm.

5. Experiments

In this section, the performance of SNAC is first evaluated on detection responses and tracklets. Then, the proposed MOT system is tested on the MOT Challenge Benchmark [37].

5.1. Evaluation of SNAC

In the MOT system, the SNAC was proposed to extract discriminative features for detection responses and tracklets instead of handcrafted methods. Discrimination and accuracy were used as the main indicators to evaluate the performance of SNAC. Meanwhile, the effects of histogram inputting and the auto-encoding constraint were also evaluated. According to the order of the system framework, the performance of SNAC was first evaluated on detection responses and then tested the SNAC on tracklets. Since current public platforms do not provide annotation data for tracklets, how to make a fair comparison is a thorny issue. Therefore, the performance of SNAC was mainly compared with different constraints and handcrafted methods. In this experiment, the training processes of SNAC were carried out with graphics processing units (GPUs).

5.1.1. SNAC for Detection Responses

During tracklet generation, an SNAC(d_i^t) was established for each detection response d_i^t to implement the explicit frame-by-frame association. Through an online learning process, SNAC(d_i^t) was able to extract features for d_i^t and D_{t+1}. Then, the similarity between d_i^t and each detection of D_{t+1} could be obtained by the Euclidean distance. Statistical discrimination and variance of SNAC(d_i^t) from these similarities can be calculated. Discrimination reflects the strength of the distinguishing ability, and variance represents the robustness. To generate the tracklet set in sliding temporal windows, each SNAC(d_i^t) was trained by an incremental learning algorithm. Indicators of discrimination and variance were computed from the overall results. Another important indicator in evaluating the SNAC is the tracklet accuracy (TA). To compute TA, tracklets were treated as the final tracking results in a time window, so the metrics of MOT [38] could be used to evaluate the accuracy of tracklets. In this case, the core indicator MOTA was equal to the TA in Equation (15):

$$TA = 1 - \frac{\sum_t (FN_t + FP_t + IDs_t)}{\sum_t GT_t} \tag{16}$$

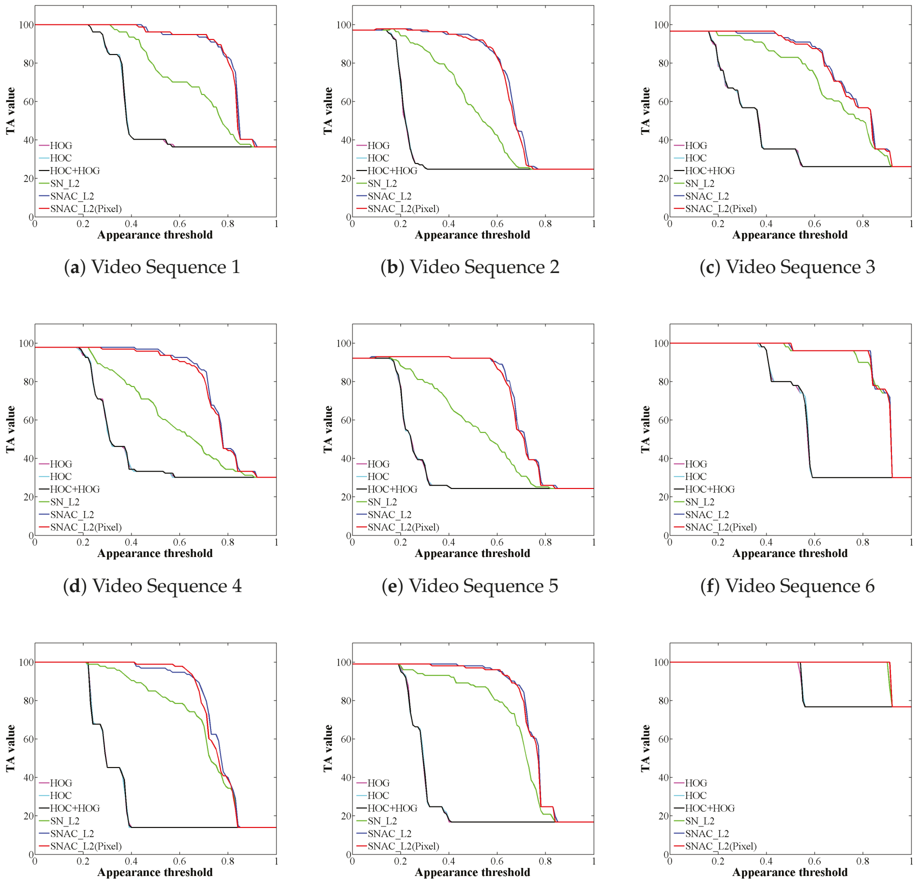
where t is the frame index in the current time window; FN, FP, and IDs are the number of false negatives, false positives, and mismatches, respectively; and GT is the number of ground truth tracklets annotated by us in this experiment.

Three subsequences of the 2DMOT2015 dataset were chosen to do this experiment. TUD-Crossing is a static camera scene, ETH-Jemoli and EHT-Linthescher are moving camera sequences. Three time windows are selected from each sequence to create a total of nine video segments for the experiment. GTs of the nine video segments are annotated.

As shown in Table 2, SNAC_L2 was chosen as the original SN with the L2 regularization constraint, the SNAC_L2(pixel) with raw pixel input, and the RGB and HOG histogram methods were used for comparison. The comparison of SNACs with traditional methods is first discussed. In Table 2, the red number in each column represents the best performance. Compared with the RGB and HOG histogram methods, the average discriminations of the SNACs were obviously superior, implying that the SNACs distinguished objects better than traditional RGB and HOG histogram methods. There were lower variances in the HOC and HOG methods due to lower discrimination. TA curves are shown in Figure 5. TA values followed the variance of the appearance threshold. From Figure 5, it can be seen that the SNACs methods were obviously better than HOC and HOG with a large threshold area. This means that SNACs were more robust. The value of TA was one when the appearance threshold was zero in these nine testing video experiments. In order to simplify the labeling works and clearly identify the relationships among objects, these nine segments were relatively simple videos with no complex interactions between objects. Thus, detections could be correctly associated only through overlapping relationships. However, it is impossible to work in a complex environment only through position and size information. Appearance is an essential factor in tracklet generation. In order to reduce the annotation workload, the experiment selected related simple scenarios. Table 2 and Figure 5 show that when a histogram was used as input, SNAC_L2 and SN_L2 were superior to the method with raw pixels as the input for all indicators. This implies that the use of the histogram as input was a more robust method that was better at suppressing detection noises. In the comparisons between SNAC_L2 and SN_L2, no significant differences in TA or average discrimination were found, but the discrimination variance of SNAC_L2 was lower. The auto-encoding constraint was shown to be useful to enhance the robustness of SNAC and made it adapt to various environments.

Table 2. Performance comparison of different features of detection responses. Red represents the best, and blue indicates the worst. HOC, histogram of color.

Methods	**SNAC_L2**		**SN_L2**		**SNAC_L2(Pixel)**		**HOC**		**HOG**		**HOC + HOG**	
Indicators	**AD**	**Var**	**AD**	**Var**	**AD**	**Var**	**AD**	**Var**	**AD**	**Var**	**AD**	**Var**
Sequence 1	0.8152	0.0568	0.8112	0.0600	0.6189	0.0989	0.1315	0.0029	0.1295	0.0030	0.1324	0.0027
Sequence 2	0.7337	0.0490	0.7387	0.0539	0.6589	0.0811	0.1494	0.0041	0.1471	0.0041	0.1503	0.0040
Sequence 3	0.7832	0.0347	0.7930	0.0362	0.7736	0.0397	0.1405	0.0017	0.1426	0.0016	0.1409	0.0017
Sequence 4	0.7983	0.0392	0.8150	0.0295	0.5807	0.0785	0.1656	0.0029	0.1657	0.0031	0.1690	0.0030
Sequence 5	0.8265	0.0194	0.8395	0.0184	0.6435	0.0759	0.1855	0.0032	0.1851	0.0036	0.1869	0.0033
Sequence 6	0.8279	0.0333	0.8232	0.0352	0.8490	0.0358	0.2043	0.0022	0.2084	0.0029	0.2061	0.0025
Sequence 7	0.7662	0.0196	0.7770	0.0280	0.7863	0.0403	0.1358	0.0015	0.1348	0.0015	0.1359	0.0015
Sequence 8	0.8149	0.0200	0.8244	0.0160	0.7813	0.0530	0.1642	0.0021	0.1651	0.0021	0.1626	0.0022
Sequence 9	0.9015	0.0001	0.9000	0.0001	0.9003	0.0001	0.1353	0.0005	0.1423	0.0003	0.1364	0.0001



(**a**) Video Sequence 1 (**b**) Video Sequence 2 (**c**) Video Sequence 3

(**d**) Video Sequence 4 (**e**) Video Sequence 5 (**f**) Video Sequence 6

(**g**) Video Sequence 7 (**h**) Video Sequence 8 (**i**) Video Sequence 9

Figure 5. Illustrations of the tracklet accuracy (TA) varying with the appearance threshold. From red to pink, they represent the SNAC_L2, SN_L2, SNAC_L2(Pixel), HOC, HOG, and HOC + HOG methods. Nine video sequences were sampled from the 2D MOT 2015 dataset and annotated. The abscissa axis indicates the appearance threshold from 0–1, and ordinates axis represents the TA up to 100. Through these curves, it can be seen that learning features are better than traditional methods at distinguishing objects in multiple object tracking (MOT). The auto-encoding constraint (AC) term and histogram inputs proposed in this paper also showed reasonable results.

5.1.2. SNAC for Tracklets

To improve the reliability of tracklet association, SNAC was improved to distinguish tracklets, and its performance is evaluated in this section. To provide fair comparisons, the average discriminations of PAN features and hand-crafted methods were evaluated. Six testing video sequences were selected from the 2D MOT 2015 dataset, and the generated tracklets in a time window were annotated for this experiment. The discrimination was calculated by the GT of tracklets, as shown in Table 3. In each sequence, there discrimination was significantly enhanced from the appearance to the composite PAN feature. Thus, PAN can effectively integrate appearance and motion to enhance discrimination.

Table 3. Discriminations of different features on tracklets.

Sequence	1	2	3	4	5	6
HOC + HOG	0.076	0.070	0.048	0.230	0.060	0.004
SNAC	0.151	0.256	0.200	0.193	0.128	0.246
PAN	0.208	0.404	0.369	0.289	0.224	0.379

5.2. Evaluation of the MOT System

In this section, the whole MOT system is evaluated using the MOT Challenge Benchmark, and the 2D MOT 2015 dataset was used for testing. Evaluation metrics are given by [38]. Multiple object tracking accuracy (MOTA) combines false positives, missed targets, and identity switches. Multiple object precision (MOTP) indicates the misalignment between GTs and tracked bounding boxes. Mostly tracked targets (MT) is the ratio of GTs that are covered by a track hypothesis for at least 80% of their respective life span. Mostly lost targets (ML) is the ratio of GTs that are covered by a track hypothesis for at most 20% of their respective life span. FP and FN are the total number of false positives and missed targets, respectively. ID switch (IDs) is the total number of identity switches. Frag is the total number of times a trajectory is fragmented.

The proposed MOT system was developed by the Theano library [39] in a Python environment. The primary station was equipped with a 4.0-GHz CPU and an NVIDIA GeForce GTX 1070 GPU.

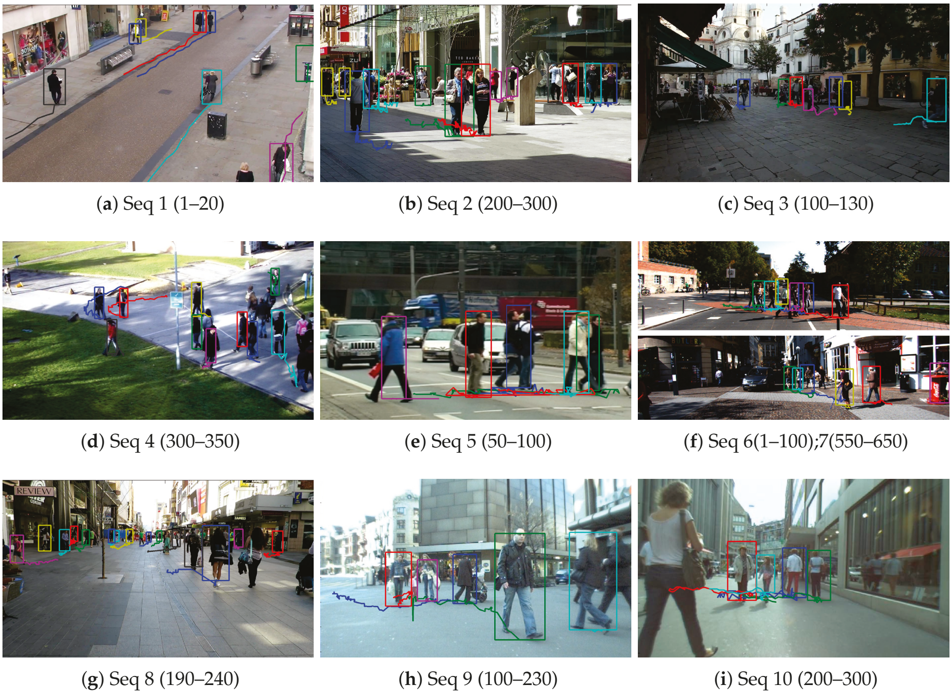
The proposed MOT system was tested on the benchmark and compared with closely related works and state-of-the-art MOT methods including those using traditional features [8,10,40,41], learning features [17,22,23,31,42,43], and higher order motion information [44]. The experimental results are listed in Table 4.

Table 4. Performance comparison of multiple object tracking (MOT) systems. Red represents the best. The upward arrow indicates the higher the better, and the downward arrow means the lower the better. MOTA, multiple object tracking accuracy; MOTP, multiple object precision; MT, mostly tracked; ML, mostly lost; Frag, the total number of times a trajectory is fragmented.

Method	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDs↓	Frag↓
Proposed	29.3	68.6	12.9%	36.3%	9880	32173	1385	2226
Siamese CNN [22]	29.0	71.2	8.5%	48.4%	5160	37,798	639	1316
HAM_INTP15 [23]	28.6	71.1	10.0%	44.0%	7485	35,910	460	1038
CEISP [40]	25.8	70.9	10.0%	44.0%	6316	37,798	1493	2240
LP_SSVM [42]	25.2	71.7	5.8%	53.0%	8369	36,932	646	849
LINF1 [41]	24.5	71.3	5.5%	64.6%	5864	40,207	298	744
TENSOR [44]	24.3	71.6	5.5%	46.6%	6644	38,582	1271	1304
DEEPDA_MOT [16]	22.5	70.9	6.4%	62.0%	7346	39,092	1159	1538
MTSTracker [43]	20.6	70.3	9.0%	63.9%	15,161	32,212	1387	2357
TC_Siamese [17]	20.2	71.1	2.6%	67.5%	6127	42,596	294	825
DCO_X [9]	19.6	71.4	5.1%	54.9%	10,652	38,232	521	819
RNN_LSTM [31]	19.0	71.0	5.5%	45.6%	11,578	36,706	1490	2081
DP_NMS [8]	14.5	70.8	6.0%	40.8%	13,171	34,814	4537	3090

The results for the MOT 2015 dataset showed that the proposed MOT system using SNAC obtained a better performance for MOTA than the other competitors listed in Table 4. The proposed method showed a comprehensive performance improvement compared with the hand-crafted feature methods CEISP and DP_NMS. This means that online learned features can better distinguish among targets and complete data association than traditional hand-crafted methods. Compared with the deep neural network feature MOT system, it can be seen that learning features is suitable for MOT applications. A higher MT indicates that tracklet growth can extend the short tracklets to enhance the PAN feature to make object trajectories as complete as possible. Meanwhile, a lower ML also benefits from the tracklet growing module. It also has disadvantages, as inaccurate detection compensation will lead to increases in FP and FN and reduce MOTP and the performance of PAN to achieve more IDs. Further improvement is needed in this area. Specific indicators such as MT and ML were superior for the proposed method than for several deep learning methods, especially the related deep Siamese network methods [17,22,23]. This implies that the online learned feature extraction method, which collects samples only from current scenes, can describe objects accurately and distinguish objects robustly. The feature extraction method with a simple structure and online training is useful for MOT. Although the proposed method was still no better than the state-of-the art methods detailed in [37], a pure online solution is possible in terms of time and performance, but this needs to be confirmed by further research.

Figure 6 demonstrates some tracking results of the proposed method on the 2D MOT 2015 dataset. For the static camera cases of Figure 6a–e and the upper part of Figure 6f, tracking results showed good performance. In Figure 6a, there are two pedestrians close in distance and alike in appearance, and they walk together. This is a difficult situation in MOT as their trajectories are likely to interfere with each other and produce false tracking results. With the help of discriminative features, the proposed method correctly tracked them. Figure 6d shows that the method can track the targets of complex movements robustly. Though scenes of the lower Figure 6f,g–i were difficult due to camera motion, the proposed method still worked properly and correctly distinguished objects.


(**a**) Seq 1 (1–20) (**b**) Seq 2 (200–300) (**c**) Seq 3 (100–130)

(**d**) Seq 4 (300–350) (**e**) Seq 5 (50–100) (**f**) Seq 6(1–100);7(550–650)

(**g**) Seq 8 (190–240) (**h**) Seq 9 (100–230) (**i**) Seq 10 (200–300)

Figure 6. Tracking results on the 2D MOT 2015 dataset. There are ten sequences in the figure, in which (**f**) contains two sequences. The ETH-Crossing sequence is not shown because it has less targets. The former six sequences (**a**–**e**) and the upper one in (**f**) are static camera cases; the rest are motion camera cases.

The execution efficiency of the proposed method is shown in Table 5. As the execution efficiency of MOT methods tested on the MOT Challenge Benchmark were not calculated officially, but uploaded by the authors themselves, it is hard to make fair comparisons. Multiple object tracking is a system including tracklet generation, tracking model establishment, tracklet association, trajectories generation, and other specific modules. The runtime performance of the main modules in the proposed MOT system are shown in Table 5, which is conducive to specific analysis. In the proposed MOT system, tracklet generation, tracklet association, and tracking results generation were executed with a 4.0-GHz CPU, and detection training and tracklet training were ran by a Nvidia Geforce GTX 1070 GPU card. From Table 5, the efficiencies of tracklet generation and trajectory generation basically met the real-time requirements. However, the training of SNAC consumed much time and reduced the efficiency of the whole MOT system. The main reason was that the program codes were encoded only for the purpose of functions evaluation and have not been optimized for running efficiency. In addition, the hardware was not an engineering-grade graphics card. Further works will be carried out for real-time implementation of the proposed MOT framework.

Table 5. Specific execution efficiency of proposed MOT system. Time consumption (C) and execution efficiency (E) of the whole MOT system and main modules are calculated.

Modules	**Detections Training**		**Tracklets Generation**		**Tracklets Training**		**Trajectories Generation**		**Whole System**	
Indicators	**C (sec)**	**E (fps)**	**C (sec)**	**E (fps)**	**C (sec)**	**E (fps)**	**C (sec)**	**E (fps)**	**C (sec)**	**E (fps)**
AVG-Town	24.1358	0.0414	0.0255	39.2311	20.2524	0.0494	0.0383	26.1271	44.4519	0.0225
ADL-1	26.9062	0.0372	0.0460	21.7297	12.0443	0.0830	0.0186	53.6481	39.0152	0.0256
Venice	12.0126	0.08328	0.0021	469.7286	3.4141	0.2929	0.0036	278.74	15.4324	0.0648
PETS2L2	28.6360	0.0349	0.0328	30.4669	27.0256	0.0370	0.0622	16.0834	55.7565	0.0179
TUD-Cro	5.1749	0.1932	0.0063	157.8591	0.9073	1.1022	0.0012	840.3361	6.0897	0.1642
KITTI16	14.7907	0.0676	0.0174	57.4918	3.6223	0.2761	0.0160	62.3750	18.4464	0.0542
KITTI19	4.1255	0.2424	0.0059	170.6446	0.6580	1.5198	0.0034	292.0029	4.7928	0.2086
ADL-3	15.1071	0.0662	0.0220	45.5284	1.2581	0.7949	0.0026	389.1656	16.3897	0.0610
ETH-Jel	5.9893	0.1670	0.0083	120.0808	0.6869	1.4559	0.0017	582.0106	6.6862	0.1496
ETH-Lin	4.9171	0.2034	0.0098	101.5885	0.6640	1.5059	0.0011	946.5673	5.5921	0.1788
ETH-Cro	3.6163	0.2765	0.0050	199.2754	0.3902	2.5631	0.0006	1657.8749	4.0121	0.2492

6. Conclusions

In this paper, an SNAC method has been presented to better distinguish objects for MOT. The online learned SNAC can work well in noisy and small sample environments. An incremental learning SNAC algorithm was proposed to generate reliable tracklets. SNAC was also improved to extract an PAN feature that combines appearance and motion for distinguishing tracklets. Tracklet growth was used to compensate for missing detections to improve the association.

Two sub-experiments were designed to evaluate the performance of SNAC and the PAN feature. The experimental results showed that SNAC could extract discriminative features from detection responses and better distinguish them. Meanwhile, in terms of appearance, PAN had a significant improvement in discrimination over SNAC and could better carry out tracklet association. The whole tracking system was evaluated over the 2D MOT 2015 dataset, and the results were compared with the state-of-the-art methods, showing a comparable performance. Experiments showed that this kind of pure online feature extraction solution is suitable for MOT.

Further research includes two aspects. One is combining more useful information to improve the proposed feature extraction method to better distinguish objects for MOT. Another is improving the efficiency of the proposed method to achieve real-time tracking.

Author Contributions: Conceptualization, P.L., X.L., and Z.F.; data curation, P.L. and H.L.; formal analysis, P.L. and X.L.; funding acquisition, X.L. and Z.F.; investigation, P.L. and H.L.; methodology, P.L. and X.L.; project administration, X.L. and Z.F.; resources, Z.F.; software, P.L. and H.L.; supervision, X.L. and Z.F.; validation, P.L. and H.L.; visualization, P.L. and H.L.; writing, original draft, P.L.; writing, review and editing, P.L. and X.L.

Funding: This research was funded by the National Natural Science Foundation of China under Grant 61671126.

Acknowledgments: The authors would like to acknowledge the Multiple Object Tracking Benchmark platform for providing fair comparative experimental data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Felzenszwalb, P.F.; Girshick, R.B.; Mcallester, D.; Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1627–1645. [CrossRef] [PubMed]
2. Dollár, P.; Appel, R.; Belongie, S.; Perona, P. Fast feature pyramids for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1532–1545. [CrossRef] [PubMed]
3. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *39*, 1137–1149. [CrossRef] [PubMed]
4. Huang, C.; Li, Y.; Nevatia, R. Multiple target tracking by learning-based hierarchical association of detection responses. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 898–910. [CrossRef] [PubMed]
5. Yang, B.; Nevatia, R. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 1918–1925.
6. Zhang, L.; Li, Y.; Nevatia, R. Global data association for multi-object tracking using network flows. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
7. Chari, V.; Lacoste-Julien, S.; Laptev, I.; Sivic, J. On pairwise costs for network flow multi-object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5537–5545.
8. Pirsiavash, H.; Ramanan, D.; Fowlkes, C.C. Globally optimal greedy algorithms for tracking a variable number of objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1201–1208.
9. Milan, A.; Schindler, K.; Roth, S. Multi-target tracking by discrete-continuous energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2054–2068. [CrossRef] [PubMed]
10. Milan, A.; Roth, S.; Schindler, K. Continuous energy minimization for multitarget tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 58–72. [CrossRef]
11. Yang, B.; Nevatia, R. Multi-target tracking by online learning a crf model of appearance and motion patterns. *Int. J. Comput. Vis.* **2014**, *107*, 203–217. [CrossRef]
12. Zhou, H.; Ouyang, W.; Cheng, J.; Wang, X.; Li, H. Deep continuous conditional random fields with asymmetric inter-object constraints for online multi-object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 58–72. [CrossRef]
13. Wen, L.; Lei, Z.; Lyu, S.; Li, S.Z.; Yang, M.H. Exploiting hierarchical dense structures on hypergraphs for multi-object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 1983–1996. [CrossRef] [PubMed]
14. Schulter, S.; Vernaza, P.; Choi, W.; Chandraker, M. Deep network flow for multi-object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2730–2739.
15. Wang, B.; Wang, L.; Shuai, B.; Zuo, Z.; Liu, T.; Chan, K.L.; Wang, G. Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26–30 June 2016; pp. 386–393.
16. Yoon, K.; Kim, D.Y.; Yoon, Y.C.; Jeon, M. Data association for multi-object tracking via deep neural networks. *Sensors* **2019**, *19*, 559. [CrossRef] [PubMed]
17. Yoon, Y.C.; Song, Y.M.; Yoon, K.; Jeon, M. Online multi-object tracking using selective deep appearance matching. In Proceedings of the IEEE Conference on Consumer Electronics-Asia, Jeju, Korea, 24–26 June 2018; pp. 206–212.
18. Dalal, V.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
19. Wang, X.; Han, T.X.; Yan, S. An hog-lbp human detector with partial occlusion handling. In Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 32–39.
20. Kuo, C.H.; Nevatia, R. How does person identity recognition help multi-person tracking? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1217–1224.

21. Bae, S.H.; Yoon, K.J. Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 595–610. [CrossRef] [PubMed]
22. Leal-Taixé, L.; Canton-Ferrer, C.; Schindler, K. Learning by tracking: Siamese cnn for robust target association. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26–30 June 2016; pp. 418–425.
23. Yoon Y.C.; Boragule, A.; Song, Y.M.; Yoon, K.; Jeon, M. Online multi-object tracking with historical appearance matching and scene adaptive detection filtering. In Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance, Auckland, New Zealand, 27–30 November 2018; pp. 1–6.
24. Wang, N.; Yeung, D.Y. Learning a deep compact image representation for visual tracking. In Proceedings of the International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 809–817.
25. Feng, H.; Li, X.; Liu, P.; Zhou, N. Using stacked auto-encoder to get feature with continuity and distinguishability in multi-object tracking. In Proceedings of the International Conference on Image and Graphics, Shanghai, China, 13–15 September 2017; pp. 351–361.
26. Kuo, C.H.; Huang, C.; Nevatia, R. Multi-target tracking by on-line learned discriminative appearance models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 685–692.
27. Butt, A.A.; Collins, R.T. Multiple target tracking using frame triplets. In Proceedings of the IEEE Conference on Asian Conference on Computer Vision, Daejeon, Korea, 5–9 November 2012; pp. 163–176.
28. Wang, L.; Ouyang, W.; Wang, X.; Lu, H. Visual tracking with fully convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3119–3127.
29. Nam, H.; Han, B. Learning multi-domain convolutional neural networks for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4293–4302.
30. Chen, X.; Zhang, X.; Tan, H.; Lan, L.; Luo, Z.; Huang, X. Multi-granularity hierarchical attention siamese network for visual tracking. In Proceedings of the 2018 International Joint Conference on Neural Networks, Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.
31. Milan, A.; Rezatofighi, S.H.; Dick, A.; Reid, I.; Schindler, K. Online multi-target tracking using recurrent neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4225–4232.
32. Sadeghian, A.; Alahi, A.; Savarese, S. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 300–311.
33. Son, J.; Baek, M.; Cho, M.; Han, B. Multi-object tracking with quadruplet convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3786–3795.
34. Hu, J.; Lu, J.; Tan, Y.P. Discriminative deep metric learning for face verification in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1875–1882.
35. Li, W.; Zhao, R.; Xiao, T.; Wang, X. Deepreid: Deep filter pairing neural network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 152–159.
36. Wen, L.; Lei, Z.; Chang, M.C.; Qi, H.; Lyu, S. Multi-camera multi-target tracking with space-time-view hyper-graph. *Int. J. Comput. Vis.* **2017**, *112*, 313–333. [CrossRef]
37. Milan, A.; Leal-Taixé, L.; Schindler, K.; Cremers, D.; Roth, S.; Reid, I. Multiple Object Tracking Benchmark. 2015. Available online: https://motchallenge.net (accessed on 10 March 2019).
38. Stiefelhagen, R.; Bernardin, K. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP J. Image Video Process.* **2008**, *2008*, 1–10.
39. Deep Learning Tutorials. 2013. Available online: http://deeplearning.net/tutorial/ (accessed on 2 October 2018).
40. Liu, P.; Li, X.; Feng, H.; Fu, Z. Multi-object tracking by virtual nodes added min-cost network flow. In Proceedings of the IEEE International Conference on Image Processing, Beijing, China, 17–20 September 2017; pp. 2577–2581.

41. Fagot-Bouquet, L.; Audigier, R.; Dhome, Y.; Lerasle, F. Improving multi-frame data association with sparse representations for robust near-online multi-object tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
42. Wang, S.; Fowlkes, C.C. Learning optimal parameters for multi-target tracking with contextual interactions. *Int. J. Comput. Vis.* **2017**, *122*, 484–501. [CrossRef]
43. Pang, Y.; Shi, X.; Jia, B.; Blasch, E.; Sheaff, C. Multiway histogram intersection for multi-target tracking. In Proceedings of the IEEE International Conference on Information Fusion, Washington, DC, USA, 6–9 July 2015; pp. 1938–1945.
44. Shi, X.; Ling, H.; Pang, Y.; Hu, W.; Chu, P.; Xing, J. Rank-1 tensor approximation for high-order association in multi-target tracking. *Int. J. Comput. Vis.* **2019**, 1–21. [CrossRef]

electronics

Article

Automatic Scene Recognition through Acoustic Classification for Behavioral Robotics

Sumair Aziz [1], Muhammad Awais [2,*], Tallha Akram [2], Umar Khan [1], Musaed Alhussein [3] and Khursheed Aurangzeb [3,*]

[1] Department of Electronic Engineering, University of Engineering and Technology Taxila, Taxila 47080, Pakistan; sumair.aziz@uettaxila.edu.pk (S.A.); umar.khan@uettaxila.edu.pk (U.K.)

[2] Department of Electrical and Computer Engineering, COMSATS University Islamabad—Wah Campus, Wah Cantt 47040, Pakistan; tallha@ciitwah.edu.pk

[3] Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; musaed@ccis.ksu.edu.sa

* Correspondence: muhammadawais@ciitwah.edu.pk (M.A.); kaurangzeb@ksu.edu.sa (K.A.); Tel.: +92-346-5316595 (M.A.)

Received: 27 March 2019; Accepted: 23 April 2019; Published: 30 April 2019

Abstract: Classification of complex acoustic scenes under real time scenarios is an active domain which has engaged several researchers lately form the machine learning community. A variety of techniques have been proposed for acoustic patterns or scene classification including natural soundscapes such as rain/thunder, and urban soundscapes such as restaurants/streets, etc. In this work, we present a framework for automatic acoustic classification for behavioral robotics. Motivated by several texture classification algorithms used in computer vision, a modified feature descriptor for sound is proposed which incorporates a combination of 1-D local ternary patterns (1D-LTP) and baseline method Mel-frequency cepstral coefficients (MFCC). The extracted feature vector is later classified using a multi-class support vector machine (SVM), which is selected as a base classifier. The proposed method is validated on two standard benchmark datasets i.e., DCASE and RWCP and achieves accuracies of 97.38% and 94.10%, respectively. A comparative analysis demonstrates that the proposed scheme performs exceptionally well compared to other feature descriptors.

Keywords: feature extraction; sound classification; support vector machine; sound processing; robotics; MFCC

1. Introduction

Robotics is the branch of artificial intelligence which is concerned with designing robots that can perform tasks and interact with the environment, without the aid of human intervention. Although the mechanical control technology of robots has been remarkably well developed in recent years. The ability of robots to perceive and analyse their surrounding environment, especially the auditory scenes still requires a significant research effort. Acoustic-based classification complements the vision based classification in a number of ways. First, considering the field of view, microphones are more nearly omni-directional than even wide-angle camera lenses. Second, audio signals require a significantly smaller bandwidth and low processing power. Third, acoustic classification is more reliable as the parameters of image/video processing algorithms are affected by variations in light intensity, thus, increasing the probability of false alarms. Detection and classification of acoustic scenes can help to facilitate the human-robot interaction and increase the application domain of behavioral and assistive robotics.

One of the key aspects of designing an acoustic classification system is the selection of proper signal features that could achieve an effective discrimination between different sound signals. Sounds coming from a general environment are considered neither music nor speech, but a collection of some audio signals that resemble noise signals. While sufficient research has focused on music and speech analysis, very little work has been done on concrete analysis of feature selection for classification of environmental sounds. One of the main objectives of this research is to investigate the effect of multiple features on the efficiency of an environmental scene classification system.

The state-of-the-art for acoustic scene classification features a number of approaches. Table 1 presents a summary of some considerable works in this domain which are discussed as follows. In [1], an approach based on local binary patterns (LBP) is adopted to construct the spectrogram image of environmental sounds. The LBP features are enhanced by incorporating local statistics, normalized and finally classified by a linear SVM. The accuracy is validated against RWCP dataset. In [2], the authors studied sound classification in a non-stationary noise environment. At first, probabilistic latent component analysis (PLCA) is performed for noise separation. Further, regularized kernel fisher discriminant analysis (KFDA) is adopted for multi-class sound classification. The method is validated on RWCP dataset. In [3], acoustic classification is performed using large-scale audio feature extraction. First, a large number of spectral, cepstral, energy and voice related features are extracted from highly variable recordings. Then, a sliding window approach is adopted with SVM to classify short recordings. Finally, a majority voting is employed to classify large recordings. The work further proposes Mel spectra as the most relevant features.

Table 1. Summary of published works on acoustic scene classification.

Work	Features	Classifier	Dataset	Accuracy
[1]	ID-LBP	Linear SVM	RWCP	98%
[2]	PLCA, temporal–spectral patterns of sound spectrogram	FDA	RWCP	91.04%
[3]	MFCC, Spectral and energy features	SVM	DCASE	73%
[4]	Multichannel LBP	SVM	RWCP, NTU-SEC	99.85%, 96.29%
[5]	Matching Pursuit and MFCC	GMM	BBC sound effects	98.4%
[6]	Thresholds based pre-processing, FFT	SVM	Self collected 250 recordings of dropping and hitting sounds	87%
[7]	LFCC	GMM	self collected dataset using a microphone set up on cleaning robot platform	90%
[8]	HOG	pooling	DCASE-challenge, Litis Rauin, EA	70%
[9]	MP decomposition using Gabor function with time frequency histogram	Random Forest	Combination of self collected sounds, Sound Idea database [10], Free sound project [11]	
[12]	Deep neural network based transfer learning	Softmax	DCASE	85.6%
[13]	MFCC	CNN	UrbanSoundK	77%
[14]	Multiple	Hierarchical	Self collected	92.6%
[15]	MFCC, ZC, LAR etc.	KNN	Self Collected	99%
[16]	average peak, height & width, no. of half-wavelengths of music wave	Regression analysis	self collected	77%

In [4], features based on LBP from the logarithm of the Gammatone-like spectrogram are proposed. However, LBP is sensitive to noise and discards important information. Therefore, a two-projection-based

LBP feature descriptor is also proposed that captures the texture information of the spectrogram of sound events. In [5], a matching pursuit (MP) algorithm is used to extract effective time-frequency features from sounds. The MP technique uses a dictionary of atoms for feature selection, resulting in a set of features that are flexible and physically interpretable. In [6], Fast Fourier Transform (FFT) is used to extract spectral power and duration of event based sounds. A number of features are extracted which include time-domain zero crossings, spectral centroid, roll off, flux and MFCC. Further, sound classification is done using SVM and multi-layer perceptron (MLP). In [7], a combination of log frequency cepstral coefficient (LFCC), Gaussian mixture models (GMMs) and a maximum likelihood criterion is employed to recognize various sound events for a cleaning robot. Experimental results demonstrate that LFCC based approach performs better than MFCC under low signal to noise ratio (SNR) environment. Human classification accuracy in performing similar classification tasks is also evaluated by experiments.

In [8], a feature extraction pipeline is proposed for analyzing audio scene signals. Features are computed from a histogram of gradients (HOG) of constant Q-transform followed by an appropriate pooling scheme. The performance of the proposed scheme is tested on several datasets including Toy, East Anglia (EA) and another dataset named Litis Rouen collected by the authors. In [9], MP algorithm is used to extract useful Gabor atoms from input audio stream. MP is applied over the whole duration of acoustic event. The time-frequency features are constructed from atoms in order to capture temporal and spectral information of a sound event. Further, the classification is done using a random forest classifier. Deep neural network (DNN) based transfer learning is proposed in [12] for acoustic classification. First, the DNN is trained on source domain task that performs mid-level feature extraction. Then, the pre-trained model is re-used on the DCASE target task. In [13], the authors proposed that dilated CNN architecture performs better environmental sound classification as compared to CNN with max pooling. The effect of dilation rate and number of layers on performance is also investigated. The work in [14] proposes a hierarchical approach to classify different sound events such as silence, non-silence, speech, non-speech, music and noise. In contrast to a classical one-step classification scheme, a different set of effective features is selected at each level. In [15], a hearing aid system is proposed for real time recognition of various sounds. The system is based on generating audio finger print i.e., a brief summary of audio file which collects a number of features including spectrogram zero crossings (ZC), MFCCs, linear prediction coefficients (LPCs) and log area ratio (LAR). The recognition is done on self collected sound samples using a K nearest neighbors (KNN) classifier. The system achieves a maximum accuracy of 99%. In [16], the authors propose automatic emotion classification system for music sounds. The work utilizes several features of sound wave, i.e., peak value, average height, the number of half wavelengths, average width and beats per minutes. Finally, regression analysis is perform to recognize various emotions from the sound. The system achieves an average accuracy of 77%. In [17], sound identification method for a mobile robot in home and office environment is proposed. A simple sound database called Pitch-Cluster-Maps (PCMs) based on vector quantization technique is constructed and its codebook is generated using binarized frequency spectrum. The works in [18,19] demonstrate that acoustic local ternary patterns (LTPs) show better performance as compared to MFCCs for fall detection problem. In the literature, various convolutional neural network (CNN) architectures are used to classify soundtracks from a dataset of 70 million training videos (5.24 million hours) with $30,871$ video-level labels [20]. Experiments are performed using fully connected DNNs, VGG [21], AlexNet [22], Inception [23] and ResNet [24] etc.

The acoustic scene classification approach proposed in this work has the following contributions.

- An extended feature descriptor is proposed which takes advantage of modified 1-D LTP in combination with MFCC.
- A feature fusion methodology is opted, which exploits the complementary strengths of both MFCC and modified 1-D features to generate a serial vector.
- To provide a better insight, a set of classifiers are tested on two standard benchmark datasets. This action supports researchers in selecting the best classifiers for this application.

The rest of the paper is organized as follows. In Section 2, the proposed method of acoustic scene classification is discussed. Section 3 discusses the experimental setup and datasets. The performance results and discussions are presented and discussed in Section 4 and finally, Section 5 concludes the paper.

2. Proposed Method

2.1. System Overview

Figure 1 shows the overall architecture of the proposed acoustic scene classification system. The sound signal is captured from environment through a microphone. It is digitized using an ADC in the preprocessing step and fed into the feature extraction stage. The MFCC and 1D-LTP features are extracted from the digital sound signal, they are fused together in a joint feature vector and finally classified using an SVM classifier. The main processing steps of the proposed system are discussed as follows.

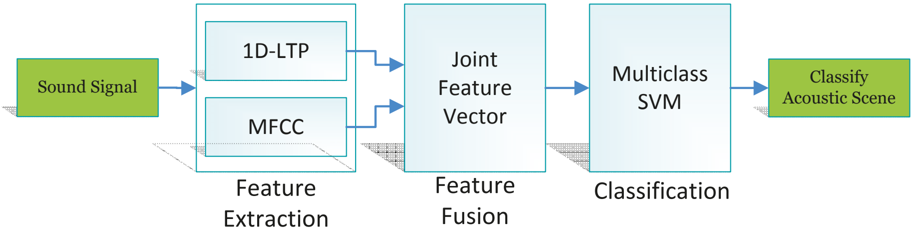


Figure 1. System Architecture for Acoustic Scene Classification.

2.2. Feature Extraction

2.2.1. 1-D Local Ternary Patterns

The local binary patterns (LBPs) have been investigated as among the most prominent feature descriptors in the field of computer vision and image analysis [25]. The basic idea behind LBP is to compare each pixel of an image with its neighborhood. Each compar ison of an image pixel with its neighbors results in binary values ’0’ or ’1’. This helps to summarize a local structure in an image and obtains powerful feature descriptors for a number of promising applications such as face recognition [26] and texture analysis [27]. LBPs are invariant to monotonic grey scale changes and have low computational cost [28]. Applying the LBP method for 1-D signals such as sound, helps to obtain useful information about local temporal dynamics of sound. The LBPs achieve discriminative features of several sounds, as exhibited by the works on music genre recognition [29] as well as environmental sound classification [1]. However, LBPs are highly affected by noise and fluctuations in acoustic samples [1]. In order to further improve the discriminative power of LBP, LTPs were proposed for face recognition in 2010 [30], and later on applied in a number of works [31–33]. In contrast to the LBPs which encode the relationships of ’greater than’ or ’less than’ between the pixel and its neighbor, the LTPs reflect the ’greater than’, ’equal to’ or ’less than' relationships. Under the same sampling conditions, LTPs help to achieve more discriminative and sophisticated sound features as compared to 1D-LBPs.

Analog audio signal is first digitized with sampling frequency F_s to form a discrete signal $X[i]$ having N number of samples. The 1D-LTPs of sampled signal $X[i]$ are computed using a sliding window approach. Consider a signal sample $x[i]$ with amplitude α is placed at the center of window with size $P+1$. Defining the upper and lower values of amplitude threshold as $(\alpha+t)$ and $(\alpha-t)$ respectively, where t is arbitrary constant. From the amplitudes of signal samples that lie in the window, a ternary code vector F of size P is obtained whose individual values are computed as;

$$F[j] = Q(x[i + \frac{P}{2} - r]),\ \forall j \in \{0, \cdots, P-1\}, \tag{1}$$

$$r = \left\{ \begin{array}{cc} j & j < \frac{P}{2} \\ j+1 & j \geq \frac{P}{2} \end{array} \right\}, \tag{2}$$

where $Q(x[i])$ is defined as;

$$Q(x[i]) = \left\{ \begin{array}{cc} 1, & x[i] > (\alpha+t) \\ 0, & (\alpha-t) \leq x[i] \leq (\alpha+t) \\ -1, & x[i] < (\alpha-t) \end{array} \right\}. \tag{3}$$

From the ternary code vector, the upper and lower local ternary patterns are computed as;

$$LTP_{upper}[i] = \sum_{k=0, k\neq i}^{P-1} s_u(F[k]) \cdot 2^k, \tag{4}$$

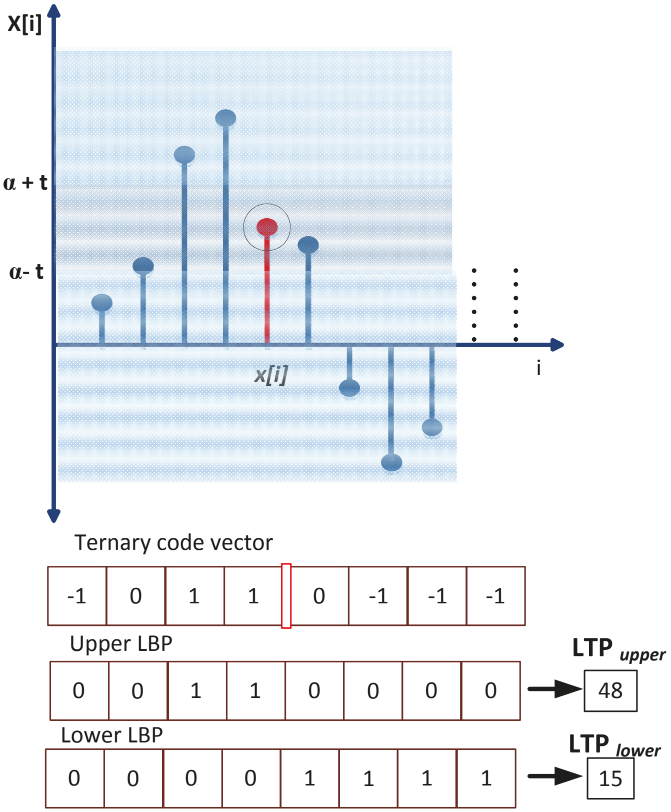
$$LTP_{lower}[i] = \sum_{k=0, k\neq i}^{P-1} s_l(F[k]) \cdot 2^k, \tag{5}$$

where,

$$s_u(F[k])) = \left\{ \begin{array}{cc} 1 & F[k] = 1 \\ 0 & otherwise \end{array} \right\}, \tag{6}$$

$$s_l(F[k])) = \left\{ \begin{array}{cc} 1 & F[k] = -1 \\ 0 & otherwise \end{array} \right\}, \tag{7}$$

Figure 2 illustrates the extraction of 1D-LTP features for one sample of a discrete audio signal.



Ternary code vector

-1	0	1	1	0	-1	-1	-1

Upper LBP

0	0	1	1	0	0	0	0

Lower LBP

0	0	0	0	1	1	1	1

Figure 2. Extraction of 1D-LTP features.

2.2.2. Mel-Frequency Cepstral Coefficients (MFCC)

MFCCs are a baseline method that has been widely used in the analysis of audio signals. Although primarily designed for speech recognition [34,35], they have been a popular feature of choice in the automatic scene classification [36,37]. The MFCCs are the coefficients that collectively make up the Mel Frequency Cepstrum (MFC), a representation of short term power spectrum of sound based on linear cosine transform of a log power spectrum on a non linear Mel scale of frequency. The MFCCs are linearly spaced on the Mel frequency scale which closely approximates the human auditory system's response. Such a representation of sound signal extracts discriminant features which help to achieve environmental sound classification with good accuracy.

Figure 3 shows a standard pipeline for the extraction of MFCC features. In the first step, the digitized sound signal is segmented in to short frames each having N samples. Next, the periodogram-based power spectrum is estimated for each frame. Let $s_i(n)$ denote the time domain signal (of N samples) that belongs to frame i, its Discrete Fourier Transform (DFT) is calculated as;

$$S_i(k) = \sum_{n=1}^{N} s_i(n)h(n)e^{-j2\pi kn/N}, \; 1 \leq k \leq K \tag{8}$$

where K denotes the length of DFT and $h(n)$ denotes the N sample long analysis window. In this work, Hamming window is used to realize a high-pass FIR filter to emphasize the high frequency part of the signal and remove DC content. In the next step, the output of complex Fourier transform is magnitude squared and power spectral estimate of frame i is computed as;

$$P_i(k) = \frac{1}{N}|S_i(k)|^2, \ 1 \leq k \leq K. \tag{9}$$

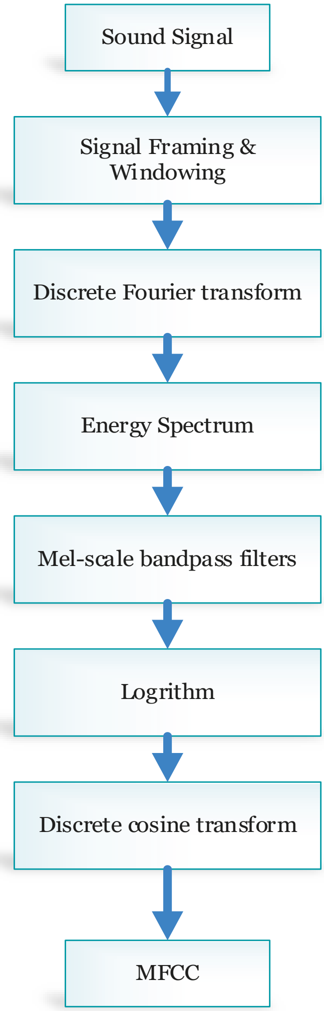

Sound Signal
Signal Framing & Windowing
Discrete Fourier transform
Energy Spectrum
Mel-scale bandpass filters
Logrithm
Discrete cosine transform
MFCC

Figure 3. MFCC Feature Extraction Pipeline.

Then, a set of Mel-scaled filter banks is computed and applied to power spectrum of each frame. The Mel-scale is linear for frequencies lower than 1000 Hz and a logarithm above it. To compute the filter bank energy spectrum, each filter is multiplied by the power spectrum computed above and coefficients are added up. The Mel-filtered spectrum of frame *i* is computed as;

$$E_i(l) = \sum_{k=0}^{N-1} P_i(k) H_l(k), \;\; \forall l = 1, \cdots, L \tag{10}$$

where L denotes the total number of filters and H_l denotes the transfer function of lth filter. Next, the *logarithm* of Mel-filtered energy spectrum is computed and Discrete Cosine Transform (DCT) is applied to it. Mathematically,

$$E_l^{'}(l) = \log(E_i(l)), \;\; \forall l = 1, \cdots, L \tag{11}$$

$$c_i(n) = \sum_{l=1}^{L} E_l^{'}(l) cos(n(l-0.5)/\pi/L) \tag{12}$$

where $n = 1, \cdots, L$ is the cepstral coefficient number. In the proposed frame work, initial 13 MFCCs are used for scene classification.

2.3. Feature Fusion

The 1D-LTP and MFCC features extracted above are fused together to form a joint feature vector for classification. The fusion of 1D-LTP and MFCC features helps to obtain a more sophisticated feature representation which has better discriminative properties as well as an accurate representation in frequency domain. The fusion process is a simple serial concatenation of 1D-LTP and MFCC feature vectors.

$$\mathbb{F}^{(c,s)} = c_\kappa || s_\kappa \tag{13}$$

2.4. Classification

The classification stage employs a multiclass SVM. The basic idea of SVM is to find a hyperplane that separates D-dimensional data into its two classes [38]. SVM is a discriminative model for classification that principally depends on two basic assumptions. First, complex classification problems can be classified through simple linear discriminative functions by transforming data into a high-dimension space. Second, the training samples for SVMs consist only of those data points that lie close to the decision surface, with the supposition that they provide the most relevant information for classification [39]. SVMs were originally proposed as binary classifiers. However, in real scenarios, data is to be classified into multiple classes. This is done by using multiclass SVM. Either a one-against-one (OAO) or one-against-all (OAA) approach can be used [40]. For acoustic scene classification setup proposed in this work, the joint feature vector extracted from previous stage is used to train the multiclass SVM OAO classifier.

3. Experiments

3.1. Setup

Experiments were performed using MATLAB 2016a software on 2.2 GHz Intel i7 processor with 8 GB RAM. The extracted features are MFCC (13 coefficients) and 1D-LTPs (13 bins) with threshold $t = 0.0002$. The classification is being done by applying various SVM kernels, and by finalizing quadratic and cubic kernels because of their best performance [41]. Training/testing percentage is fixed to be 80/20 (80% for training, and 20% for testing) for both datasets. The performance of classifier is measured through classification accuracy averaged over k-fold cross validation. The value of $k = 10$ has been selected based on experimentation to generally result in best accuracy with low bias, modest variance and low correlation. The classifier accuracy is measured as,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{14}$$

where TP stands for true positive, TN for true negative, FP for false positive and FN for false negative. The performance of the proposed approach is also compared with several state-of-the-art audio feature representation techniques i.e., MFCC, ID-LBP and linear prediction cepstral coefficients (LPCC).

3.2. Datasets

An important challenge in acoustic scene classification for robotics is the collection of proper environmental sound database. Since there is an infinite number of sounds, no single database can cover all of them. Therefore, no robotic system is capable of recognizing all the sounds. Instead, the scene recognition capability is limited by the application domain and set of tasks performed by the particular robot. In order to have an initial reference for comparison, two standard benchmark datasets are selected, i.e., (a) real world computing partnership (RWCP) sound scene dataset [42] and (b) DCASE challenge dataset [43].

RWCP is one of the first datasets which are collected for scene understanding. It contains sounds of various audio sources which were moved using a mechanical device. Recordings were done using a linear array of 14 microphones and a semi-spherical array of 54 microphones with a DAT recorder at 48 KHz frequency and 16 bit resolution. The average length of sound sample is about 1 s. A proposed feature descriptor was tested on experimental dataset consisting of 17 different environmental sounds shown in Table 2 (a) along with the number of samples for each class.

The DCASE challenge dataset consists of a set of recorded sounds in fifteen different urban environments. The duration of each sound clip is 30 s and recording is performed in London. The DCASE dataset consists of 15 different classes of urban sounds; each class contains 78 sound samples as given in Table 2 (b). The RWCP and DCASE databases contain a variety of sound classes that accurately model the general indoor or outdoor environment. We believe that verifying the performance of our proposed solution on these databases can help to realize intelligent systems for advanced applications such as sound localization [44] and human–robot interaction [45,46].

As discussed earlier, 1D-LTP features are discriminative. The scatter plots of Figures 4 and 5 show the distribution of 1D-LTPs for several classes of RWCP and DCASE datasets. These plots demonstrate that the 1D-LTP feature values that belong to the same class are spaced close to each other, whereas the features belonging to different classes are spaced relatively far on the scatter plot. Features having these strong discriminative properties result in a good classification accuracy.

Table 2. Details of Individual Classes of RWCP and DCASE Datasets.

(a) RWCP Dataset

Class	No. of Samples
Aircap	100
Bells	400
Bottle	200
Buzzer	100
Case	300
Clap	400

Table 2. *Cont.*

Cup	200
Drum	100
Phone	200
Pump	100
Saw	200
Spray	100
Stapler	100
Tear	100
Toy	200
Whistle	300
Wood	300
Total	**3400**
(b) DCASE Dataset	
Class	**No. of Samples**
Beach	78
Bus	78
Cafe	78
Car	78
City Center	78
Forest	78
Grocery Store	78
Home	78
Library	78
Metro Station	78
Office	78
Park	78
Residential area	78
Train	78
Tram	78
Total	**1170**

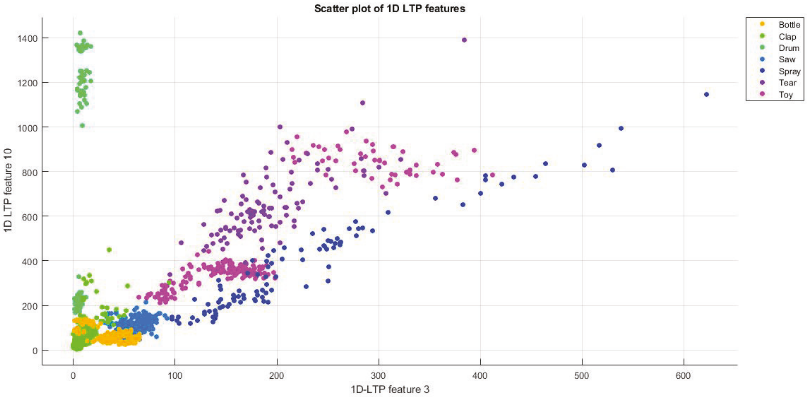


Figure 4. Scatter plot of ID-LTPs of RWCP dataset.

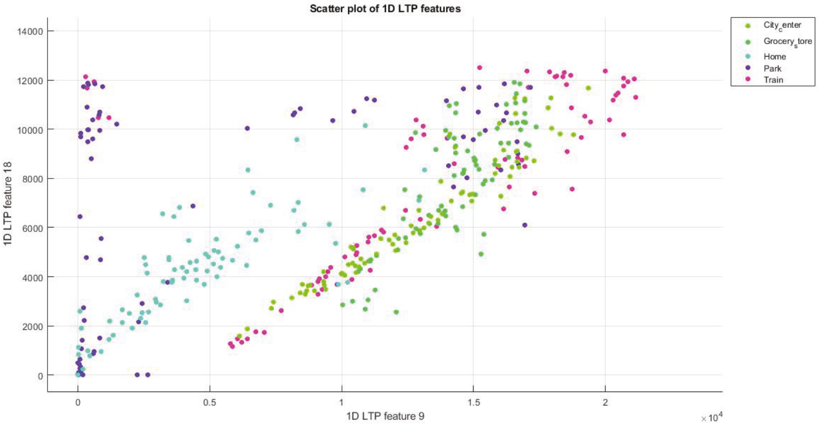


Figure 5. Scatter plot of ID-LTPs of DCASE dataset.

4. Results and Discussion

The accuracy trend for both datasets is demonstrated in Figure 6. Table 3 presents the overall classification accuracy of the proposed and existing methods along with their computational time in seconds. It can be comfortably observed from the stats that the proposed method (i.e., ID-LTP + MFCC) outperforms shows a better accuracy with computational time smaller or comparable to other approaches.

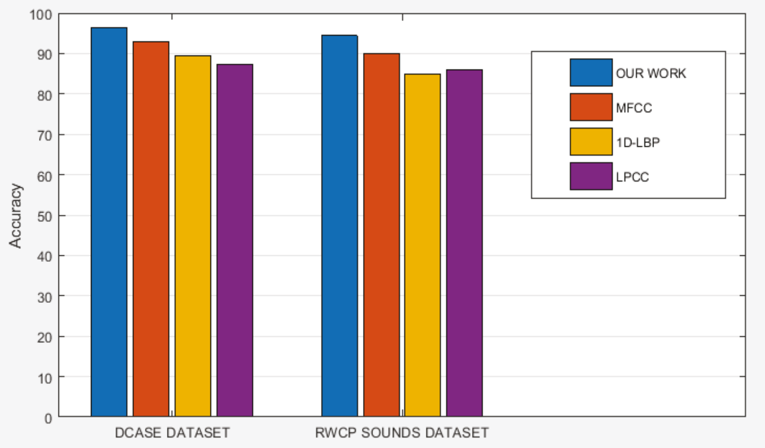


Figure 6. Classification performance of the proposed ID-LTP and several other features over DCASE and RWCP dataset.

To get a better insight, few other performance metrics are also investigated including sensitivity, specificity, and error rate. Moreover, for a fair comparison, two classifier families, i.e., SVM and KNN are contemplated due to their greater number of variants. Table 4 provides a comparison of seven classifiers on the DCASE dataset. The SVM with quadratic kernel (SVM-Q) shows better results in terms of accuracy, specificity and error rate while SVM with cubic kernel (SVM-C) and KNN weighted (KNN-W) show better sensitivity. In Table 5, the performance results are demonstrated for RWCP dataset. The SVM-Q classifier achieves a high accuracy and error rate while better sensitivity and specificity values are achieved by the KNN medium (KNN-M) and SVM-C, respectively.

Table 3. Performance results for DCASE and RWCP datasets.

Feature Descriptor	Accuracy		Time (s)
	DCASE Dataset	RWCP Sound Dataset	
MFCC	92.9%	90%	1.2
ID-LBP	89.5%	85%	0.75
LPCC	87.3%	86%	0.92
ID-LTP + MFCC	**97.38**%	**94.10**%	**0.81**

Table 4. Performance of various classifiers for proposed feature extraction approach for DCASE dataset.

DCASE Dataset				
Classifier	Performance			
	Accuracy (%)	Sensitivity (%)	Specificity (%)	Error Rate
SVM-L	89.49	83.33	99.54	0.1051
SVM-Q	**94.10**	**91.03**	**99.91**	**0.0590**
SVM-C	93.85	93.59	99.91	0.0615
SVM-G	93.16	92.31	99.82	0.0684
KNN-M	85.04	92.31	98.81	0.1496
KNN-W	90.26	93.59	99.36	0.0974
KNN-C	82.56	84.62	98.35	0.1744

Table 5. Performance of various classifiers for proposed feature extraction approach for RWCP dataset.

RWCP Dataset				
Classifier	Performance			
	Accuracy (%)	Sensitivity (%)	Specificity (%)	Error Rate
SVM-L	93.97	98.50	99.93	0.0603
SVM-Q	**97.38**	**99.0**	**99.83**	**0.0262**
SVM-C	97.26	99.25	99.97	0.0274
SVM-G	94.44	98.75	99.57	0.0556
KNN-M	97.26	99.50	99.83	0.0274
KNN-W	96.85	99.00	99.80	0.0315
KNN-C	96.35	99.25	99.80	0.0365

Classification results of individual classes for the DCASE dataset are shown by a confusion matrix of Figure 7. The figure shows that all classes except the *city center* class have an accuracy of more than 90%. The confusion matrix of the proposed approach for RWCP dataset is shown in Figure 8. Here, the *phone* class has an accuracy of 89% whereas, all the remaining classes have accuracy above 90%. The classification results of Figure 7 and 8 confirm the accuracy and validity of the proposed feature classification technique. To reveal the authenticity and robustness of our proposed method, confidence intervals against both datasets are also provided for two state-of-the-art classifiers. Figure 9 demonstrates the confidence interval showing min, max and average classification values of both classifiers. From the stats, its quite obvious that SVM-Q can be formally selected as a standard classifier for this application.

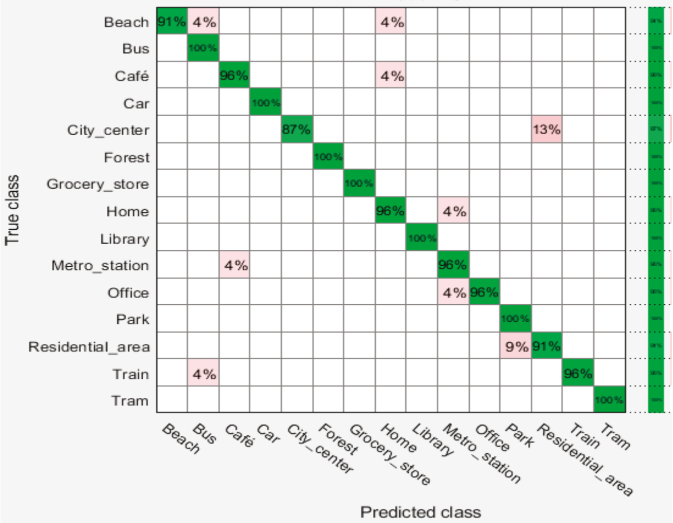


Figure 7. Confusion matrix of the proposed approach for DCASE dataset.

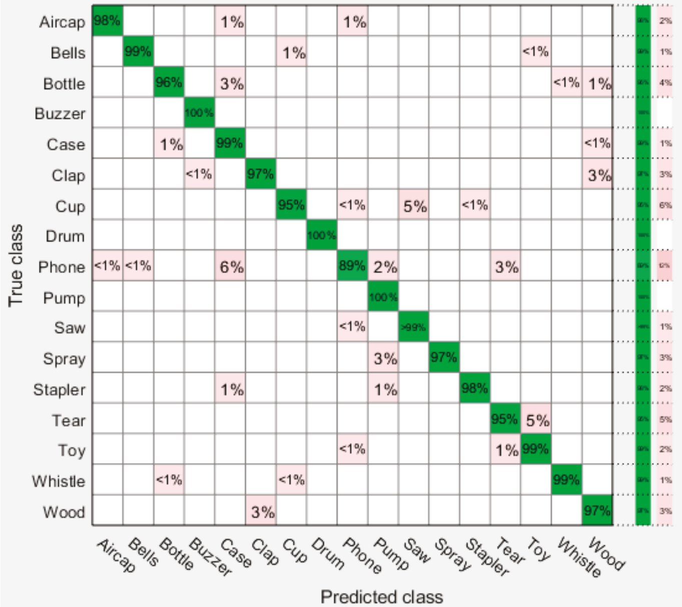


Figure 8. Confusion matrix of the proposed approach for RWCP dataset.

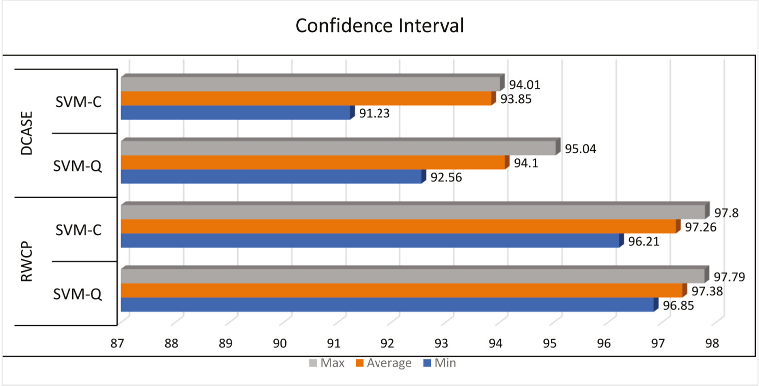


Figure 9. Confidence interval against two selected classifiers on benchmark datasets.

5. Conclusions

Scene classification is an important task in behavioral robotics. Using acoustic signals for environmental scene classification complements the visual-based classification in many ways. This study aimed to select the image texture classification features and investigate their effect on the classification of sound signals. In particular, the work proposes a modified feature descriptor as a combination of 1D-LTPs and MFCCs. Our analysis and simulation results for the two reference datasets i.e., DCASE and RWCP show that 1D-LTPs exhibit good discriminative properties for sound signals. On the other hand, the MFCCs as the baseline method, approximates the behavior of the human auditory system. Fusing 1D-LTPs with MFCCs achieves a more sophisticated and discriminative feature representation of environmental sounds. The proposed fused feature vector is classified with various kernels of multi-class SVM. Results demonstrate that SVM with quadratic kernel achieves high accuracy as compared to other feature representations. The proposed system can be applied to a number of practical indoor and outdoor robotic scenarios.

6. Materials

Two publicly available datasets are utilized in this research are RWCP and DCASE. The RWCP dataset is available at [42] and DCASE is available at: http://dcase.community/challenge2018/index.

Author Contributions: Conceptualization, S.A.; Data curation, M.A. (Muhammad Awais) and T.A.; Funding acquisition, M.A. (Musaed Alhussein) and K.A.; Investigation, M.A. (Muhammad Awais) and U.K.; Methodology, S.A., M.A. (Muhammad Awais) and U.K.; Project administration, T.A.; Resources, S.A. and K.A.; Software, M.A. (Muhammad Awais); Validation, M.A. (Musaed Alhussein) and K.A.; Writing—original draft, M.A. (Muhammad Awais).

Funding: The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group NO (RG-1438-034).

Acknowledgments: We greatly acknowledge Department of Electrical & Computer Engineering, COMSATS University Islamabad, Wah Campus and Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh for their technical support throughout the duration of this research work.

Conflicts of Interest: The authors declare that they have no competing interests.

Abbreviations

The following abbreviations are used in this manuscript:

LBP	Local Binary Patterns
LTP	Local Ternary Patterns
MFCC	Mel Frequency Cepstral Coefficients
SVM	Support Vector Machine
PCLA	Probabilistic Component Latent Analysis
KFDA	Kernel Fisher Discriminant Analysis
HOG	Histogram of Gradients
DNN	Deep Neural Networks
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
GMM	Gaussian Mixture Model
KNN	K-Nearest Neighbour
SVM-C	SVM with Cubic kernel
SVM-Q	SVM with Quadratic kernel
SVM-G	SVM with mean Gaussian kernel
KNN-M	K Nearest Neighbors-Medium
KNN-W	K Nearest Neighbors-Weighted
KNN-C	K Nearest Neighbors-Cubic
OAO	One Against One
OAA	One Against All

References

1. Kobayashi, T.; Ye, J. Acoustic feature extraction by statistics based local binary pattern for environmental sound classification. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 3052–3056.
2. Ye, J.; Kobayashi, T.; Murakawa, M.; Higuchi, T. Robust acoustic feature extraction for sound classification based on noise reduction. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 5944–5948.
3. Geiger, J.T.; Schuller, B.; Rigoll, G. Large-scale audio feature extraction and SVM for acoustic scene classification. In Proceedings of the 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, NY, USA, 20–23 October 2013; pp. 1–4.
4. Ren, J.; Jiang, X.; Yuan, J.; Magnenat-Thalmann, N. Sound-Event Classification Using Robust Texture Features for Robot Hearing. *IEEE Trans. Multimed.* **2017**, *19*, 447–458. [CrossRef]
5. Chu, S.; Narayanan, S.; Kuo, C.J. Environmental Sound Recognition With Time-Frequency Audio Features. *IEEE Trans. Audio Speech Lang. Process.* **2009**, *17*, 1142–1158. [CrossRef]
6. Saltali, I.; Sariel, S.; Ince, G. Scene Analysis Through Auditory Event Monitoring. In Proceedings of the International Workshop on Social Learning and Multimodal Interaction for Designing Artificial Agents, Tokyo, Japan, 16 November 2016; pp. 5:1–5:6.
7. Park, S.; Rho, J.; Shin, M.; Han, D.K.; Ko, H. Acoustic feature extraction for robust event recognition on cleaning robot platform. In Proceedings of the 2014 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 10–13 January 2014; pp. 145–146.
8. Rakotomamonjy, A.; Gasso, G. Histogram of Gradients of Time–Frequency Representations for Audio Scene Classification. *IEEE-ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 142–153.
9. Nguyen, Q.; Choi, J. Matching pursuit based robust acoustic event classification for surveillance systems. *Comput. Electr. Eng.* **2017**, *57*, 43–54. [CrossRef]

10. Sehili, M.A.; Lecouteux, B.; Vacher, M.; Portet, F.; Istrate, D.; Dorizzi, B.; Boudy, J. Sound Environment Analysis in Smart Home. In *Ambient Intelligence*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 208–223.
11. Wang, J.; Lin, C.; Chen, B.; Tsai, M. Gabor-Based Nonuniform Scale-Frequency Map for Environmental Sound Classification in Home Automation. *IEEE Trans. Autom. Sci. Eng.* **2014**, *11*, 607–613. [CrossRef]
12. Mun, S.; Shon, S.; Kim, W.; Han, D.K.; Ko, H. Deep Neural Network based learning and transferring mid-level audio features for acoustic scene classification. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 796–800.
13. Chen, Y.; Guo, Q.; Liang, X.; Wang, J.; Qian, Y. Environmental sound classification with dilated convolutions. *Appl. Acoust.* **2019**, *148*, 123–132. [CrossRef]
14. Saki, F.; Kehtarnavaz, N. Real-time hierarchical classification of sound signals for hearing improvement devices. *Appl. Acoust.* **2018**, *132*, 26–32. [CrossRef]
15. Yağanoğlu, M.; Köse, C. Real-Time Detection of Important Sounds with a Wearable Vibration Based Device for Hearing-Impaired People. *Electronics* **2018**, *7*, 50. [CrossRef]
16. Seo, Y.S.; Huh, J.H. Automatic Emotion-Based Music Classification for Supporting Intelligent IoT Applications. *Electronics* **2019**, *8*, 164. [CrossRef]
17. Sasaki, Y.; Kaneyoshi, M.; Kagami, S.; Mizoguchi, H.; Enomoto, T. Daily sound recognition using Pitch-Cluster-Maps for mobile robot audition. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 2724–2729.
18. Irtaza, A.; Adnan, S.M.; Aziz, S.; Javed, A.; Ullah, M.O.; Mahmood, M.T. A framework for fall detection of elderly people by analyzing environmental sounds through acoustic local ternary patterns. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 1558–1563.
19. Adnan, S.M.; Irtaza, A.; Aziz, S.; Ullah, M.O.; Javed, A.; Mahmood, M.T. Fall detection through acoustic Local Ternary Patterns. *Appl. Acoust.* **2018**, *140*, 296–300. [CrossRef]
20. Hershey, S.; Chaudhuri, S.; Ellis, D.P.W.; Gemmeke, J.F.; Jansen, A.; Moore, R.C.; Plakal, M.; Platt, D.; Saurous, R.A.; Seybold, B.; et al. CNN architectures for large-scale audio classification. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 131–135.
21. Karen, S.; Andrew, Z. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–8 December 2012.
23. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Ojala, T.; Pietikainen, M.; Harwood, D. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognit.* **1996**, *29*, 51–59. [CrossRef]
26. Zhang, B.; Gao, Y.; Zhao, S.; Liu, J. Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. *IEEE Trans. Image Process.* **2010**, *19*, 533–544. [CrossRef] [PubMed]
27. Liu, L.; Fieguth, P.; Guo, Y.; Wang, X.; Pietikäinen, M. Local binary features for texture classification: Taxonomy and experimental study. *Pattern Recognit.* **2017**, *62*, 135–160. [CrossRef]
28. Thwe, K.Z. Sound event classification using bidirectional local binary pattern. In Proceedings of the 2017 International Conference on Signal Processing and Communication (ICSPC), Tamil Nadu, India, 28–29 July 2017; pp. 501–504. [CrossRef]

29. Costa, Y.M.; Oliveira, L.; Koerich, A.L.; Gouyon, F.; Martins, J. Music genre classification using LBP textural features. *Signal Process.* **2012**, *92*, 2723–2737. [CrossRef]
30. Tan, X.; Triggs, W. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans. Image Process.* **2010**, *19*, 1635–1650. [PubMed]
31. Raja, M.; Sadasivam, V. Optimized local ternary patterns: A new texture model with set of optimal patterns for texture analysis. *J. Comput. Sci.* **2013**, *9*, 1–15. [CrossRef]
32. Wu, S.; Yang, L.; Xu, W.; Zheng, J.; Li, Z.; Fang, Z. A mutual local-ternary-pattern based method for aligning differently exposed images. *Comput. Vis. Image Underst.* **2016**, *152*, 67–78. [CrossRef]
33. Zhang, Y.; Li, S.; Wang, S.; Shi, Y.Q. Revealing the traces of median filtering using high-order local ternary patterns. *IEEE Signal Process. Lett.* **2014**, *21*, 275–279. [CrossRef]
34. Han, W.; Chan, C.F.; Choy, C.S.; Pun, K.P. An efficient MFCC extraction method in speech recognition. In Proceedings of the 2006 IEEE International Symposium on Circuits and Systems, Island of Kos, Greece, 21–24 May 2006.
35. Ittichaichareon, C.; Suksri, S. Speech Recognition using MFCC. In Proceedings of the International Conference on Computer Graphics, Simulation and Modeling (ICGSM'2012), Pattaya, Thailand, 28–29 July 2012; pp. 28–29.
36. Mesaros, A.; Heittola, T.; Virtanen, T. TUT database for acoustic scene classification and sound event detection. In Proceedings of the 2016 24th European Signal Processing Conference (EUSIPCO), Budapest, Hungary, 29 August–2 September 2016; pp. 1128–1132.
37. Shaukat, A.; Ahsan, M.; Hassan, A.; Riaz, F. Daily sound recognition for elderly people using ensemble methods. In Proceedings of the 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Xiamen, China, 19–21 August 2014; pp. 418–423.
38. Amarappa, S.; Sathyanarayana, S. Data classification using Support vector Machine (SVM), a simplified approach. *Int. J. Electron. Comput. Sci. Eng.* **2014**, *3*, 435–445.
39. Faziludeen, S.; Sabiq, P.V. ECG beat classification using wavelets and SVM. In Proceedings of the 2013 IEEE Conference on Information Communication Technologies, Thuckalay, India, 11–12 April 2013; pp. 815–818.
40. Jonathan, M.; Mohamed, C.; Robert, S. "One Against One" or "One Against All": Which One is Better for Handwriting Recognition with SVMs? In Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule, France, 23–26 October 2006.
41. Lee, S.W.; Verri, A. (Eds.) *Pattern Recognition with Support Vector Machines*; Springer: Berlin, Germany, 2002.
42. Nakamura, S.; Hiyane, K.; Asano, F.; Nishiura, T.; Yamada, T. Acoustical Sound Database in Real Environments for Sound Scene Understanding and Hands-Free Speech Recognition. In Proceedings of the 2nd International Conference on Language Resources and Evaluation, Athens, Greece, 31 May–2 June 2000.
43. Giannoulis, D.; Stowell, D.; Benetos, E.; Rossignol, M.; Lagrange, M.; Plumbley, M.D. A database and challenge for acoustic scene classification and event detection. In Proceedings of the 21st European Signal Processing Conference (EUSIPCO 2013), Marrakech, Morocco, 9–13 September 2013; pp. 1–5.
44. Rascon, C.; Meza, I. Localization of sound sources in robotics: A review. *Robot. Auton. Syst.* **2017**, *96*, 184–210. [CrossRef]
45. Toyoda, Y.; Huang, J.; Ding, S.; Liu, Y. Environmental sound recognition by multilayered neural networks. In Proceedings of the Fourth International Conference on Computer and Information Technology, Wuhan, China, 16 September 2004; pp. 123–127. [CrossRef]
46. Yamakawa, N.; Takahashi, T.; Kitahara, T.; Ogata, T.; Okuno, H.G. Environmental sound recognition for robot audition using matching-pursuit. In Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Syracuse, NY, USA, 29 June–1 July 2011; pp. 1–10.

electronics

MDPI

Article

Three-Stream Convolutional Neural Network with Squeeze-and-Excitation Block for Near-Infrared Facial Expression Recognition

Ying Chen [1,2], Zhihao Zhang [1,2], Lei Zhong [1,2], Tong Chen [1,2,3,*], Juxiang Chen [1,2] and Yeda Yu [1,2]

[1] Chongqing Key Laboratory of Nonlinear Circuit and Intelligent Information Processing, Southwest University, Chongqing 400715, China; chenyingly@email.swu.edu.cn (Y.C.); zzh085517@email.swu.edu.cn (Z.Z.); zl030610@email.swu.edu.cn (L.Z.); chenjuxiang@email.swu.edu.cn (J.C.); devil510@email.swu.edu.cn (Y.Y.)

[2] Chongqing Key Laboratory of Artificial Intelligence and Service Robot Control Technology, Chongqing Institute of Green and Intelligent Technology, CAS, Chongqing 400715, China

[3] Institute of Psychology, Chinese Academy of Sciences, Beijing 100101, China

* Correspondence: c_tong@swu.edu.cn; Tel.: +86-236-825-039

Received: 22 February 2019; Accepted: 26 March 2019; Published: 29 March 2019

Abstract: Near-infrared (NIR) facial expression recognition is resistant to illumination change. In this paper, we propose a three-stream three-dimensional convolution neural network with a squeeze-and-excitation (SE) block for NIR facial expression recognition. We fed each stream with different local regions, namely the eyes, nose, and mouth. By using an SE block, the network automatically allocated weights to different local features to further improve recognition accuracy. The experimental results on the Oulu-CASIA NIR facial expression database showed that the proposed method has a higher recognition rate than some state-of-the-art algorithms.

Keywords: NIR facial expression recognition; SE block; 3D CNN; adaptive feature weights calibration

1. Introduction

Facial expressions carry rich non-verbal information. Machines with the ability to understand facial expressions can better serve humans and fundamentally change the relationship between humans and machines. Therefore, automatic facial expression recognition has attracted attention from many fields, such as virtual reality [1,2], public security [3,4], and data-driven animation [5,6].

The effectiveness of facial expression recognition can be easily affected by environmental changes, such as changes of light, angle, and distance. Among these, the change of illumination conditions under visible light (VIS) (380–750 nm) has the largest influence [7,8]. To overcome this influence, an active near-infrared (NIR) illumination source (780–1100 nm) is used for the recognition. In this study, an NIR camera, together with the NIR illumination sources, were placed in front of the subjects. The intensity of the NIR illumination source was much higher than that of the ambient NIR light in indoor environments. Therefore, the ambient illumination problem could be solved as long as the active NIR illumination source is constant. The NIR recognition system is resistant to ambient illumination variations, and has been successfully applied to the field of face recognition [9]; it can perform well even in dark environments [10], in which normal imaging systems fail to perform recognition.

Facial expressions manifest themselves as movements of one or several discrete parts of the face, such as tightening the lips to express anger and raising the mouth to express happiness [11]. Some researchers use the features extracted from the entire face, which are called global features [12,13], for recognition, while other researchers use features extracted from specific parts, which are called local features [14–17]. Many researchers have demonstrated that local features improve the performance of facial expression recognition compared with global features [18,19]. The main reason for this advancement is that the specific local regions contribute more accurate information of facial changes that help to distinguish the expressions, while the global region contains more identity information. Some researchers [20,21] have pointed out that the eyes, eyebrows, and mouth are the most expressive facial parts. However, it is unknown which part of the face should carry more weight in expression recognition or how the correct weight can be allocated to different parts of the face.

In earlier studies, many facial expression recognition systems used static images [22–24] that only contain spatial information as the input. However, facial expression can be a dynamic process, and the dynamic information of the face can better reflect the change of expression. Therefore, it is necessary to extract spatial and temporal information from the image sequences to facilitate recognition.

In the work reported in this paper, we designed a convolutional neural network (CNN) to complete NIR facial expression recognition. The CNN used is a three-stream three-dimensional (3D) CNN, which can learn spatio-temporal information from image sequences. In addition, the three inputs to the CNN are all local features, which not only reduce computational complexity, but also remove information not related to the expressions (such as identity information). A squeeze-and-excitation (SE) block is appended after the 3D CNN, which can automatically assign more weight to the local features that carry more expression information. To overcome the over-fitting problem caused by small data, features are extracted through three identical shallow networks. Finally, we add a global face stream to the local network, further increasing the recognition rate.

The main contributions of this paper are the following: (1) Three local regions of the face are used as the input of the network for the NIR expression recognition, which can not only accurately extract the facial expression information, but also reduce the computational complexity and dimensions; and (2) an SE block is added to model the dependencies between feature channels and adaptively learn the weight of the channel to gain efficient expression information and attenuate the useless information.

2. Related Work

Facial expressions can be decomposed into movement of one or more discrete facial action units (AUs). Inspired by this theory, Liu et al. [25] located common patches and unique patches of different expressions for recognition. However, this method could cause overlapping of located areas. Liu et al. [26] did further work and proposed a framework called FDM to select the active features of each expression without overlapping. Later, Liu et al. [27] proposed a 3D CNN with deformable action part constraints that can locate and code action units.

To extract temporal features while acquiring spatial features, Ji et al. [28] extended a CNN to a 3D CNN, which can extract the spatio-temporal information from image sequences. Szegedy et al. [29] utilized the 3D CNN to extract temporal information for video-based expression recognition. Chen et al. [30] proposed a new descriptor, the histogram of oriented gradients from three orthogonal planes (HOG-TOP), to extract the dynamic texture features from image sequences, which are fused with the geometric features to identify expressions. Fonnegra et al. [31] proposed a deep learning model and Yan et al. [32] presented collaborative-discriminative-multi-metric-learning (CDMML)-based image sequences for emotion recognition. To make the system more precise, Zia et al. [33] proposed a dynamic weight majority voting mechanism for the construction of ensemble systems. However, since these methods are all based on visible light, the impact of external illumination changes are not considered.

The NIR facial images/videos are hardly influenced by the ambient visible light change. Farokhi et al. [34] proposed a method of extracting global and local features by using Zernike moments (ZMs) and Hermite kernels (HKs), respectively, and then used the fused features to identify the NIR face. Taini et al. [35] assembled a near-infrared facial expression database and completed the first study based on NIR facial expression recognition. Zhao et al. [18] developed the database of NIR facial expressions, called the Oulu-CASIA NIR facial expression database, and used local binary patterns form three orthogonal planes (LBP-TOP) to extract dynamic local features. It was proved in this work that NIR can overcome the influence of visible-light illumination changes on expression recognition. However, these methods must extract facial expression features manually. Jeni et al. [36] proposed a 3D-shape-information-based recognition technique and further proved that an NIR camera configuration is suitable for facial expressions under light-changing conditions. Wu et al. [37] proposed a three-stream 3D convolutional network for NIR facial expression recognition, using a combination of global and local features, but did not consider assigning different weights to local features.

3. Materials and Methods

3.1. 3D CNN

A 3D CNN is more suitable for spatial-temporal feature extraction. In [28], to process image sequences more efficiently, a 3D CNN approach is proposed to address action recognition problems. Through 3D convolution and pooling operations, a 3D CNN has the ability to learn temporal features.

A 3D CNN consists of an input layer, 3D convolution, 3D pooling (usually, each convolution layer is followed by the pooling layer), and a fully connected (FC) layer. The dimension of the input image sequences to the 3D CNN is represented as d × l × h × w, where d is the number of the channels, l the number of frames of video clips, and h and w the height and width, respectively, of each frame. In addition, 3D convolution and pooling have a kernel size in t × k × k, where t is the temporal depth and k the spatial size.

3.2. Squeeze-and-Excitation Networks (SENets)

Hu et al. [38] proposed squeeze-and-excitation networks (SENets). The basic architectural unit of SENets is the SE building block, which is shown in Figure 1.

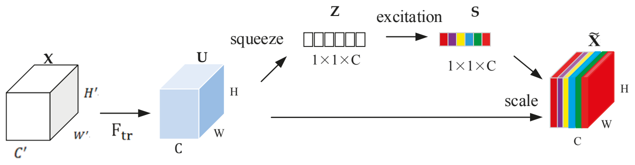


Figure 1. Squeeze-and-excitation (SE) block structure.

Before the SE block operation, input data X is transformed into features U through a series of convolution operations, i.e., $\mathrm{F}_{tr} : \mathrm{X} \rightarrow \mathrm{U}$, $\mathrm{X} \in R^{W' \times H' \times C'}$, $\mathrm{U} \in R^{W \times H \times C}$, where F_{tr} represents the transformation from X to U, H (H') and W (W') are the frame height and width, respectively, and C (C') are the number channels.

The SE block mainly consists of two operations: Squeeze and excitation. Because the filter learned by each channel in the CNN operates on the local receptive field, each feature map in U cannot utilize the context information of other feature maps. The purpose of the squeeze operation is to have a global receptive field, so that the lower layers of the network can also use global information. The global average pooling operation is used to compress U (multiple feature maps) into Z, so that the C feature maps eventually become real columns of $1 \times 1 \times C$. The squeeze operation is performed by

$$z_m = \mathrm{F}_{sq}(\mathrm{u}_m) = \frac{1}{\mathrm{W} \times H} \sum_{\mathrm{i}=1}^{W} \sum_{\mathrm{j}=1}^{H} u_{m(i,j)} \tag{1}$$

where z_m represents the mth element of Z and u_m the mth element of U.

The excitation operation is a simple gating with a sigmoid activation. The purpose of this operation is to model the interdependence between feature channels by learning parameters to generate the weight of each feature channel. To meet these requirements and limit the model complexity and auxiliary generalization, two FC layers (1*1 conv layer) were introduced. One is the dimension reduction layer, in which the parameter is W_1 and the dimension reduction ratio r; the other is a dimension increase layer with parameter W_2 followed by a Rectified linear unit (ReLU), $W_1 \in R^{\frac{C}{r}\times C}$ and $W_2 \in R^{C\times\frac{C}{r}}$. The excitation is performed by:

$$S = F_{ex}(Z,\ W) = \sigma(g(Z,\ W)) = \sigma(W_2\,\delta(W_1, Z)) \tag{2}$$

where S is the vector after excitation operation, and δ and σ refer to the ReLU function and the sigmoid function, respectively.

Finally, S is combined with U to obtain the final output by:

$$\tilde{x}_m = F_{scale}(u_m,\ s_m) = s_m \cdot u_m \tag{3}$$

where s_m is the mth element of S and $\tilde{x}_m$ the mth element of the final output $\tilde{X}$; F_{scale} refers to channel-wise multiplication.

The goal of the SE block is to greatly improve the expressiveness of the network; it adaptively recalibrates the feature weight by modeling the interdependencies between the channels. In more detail, it allows the network to use global information to selectively enhance the beneficial features of the channel and suppress the useless function channels.

3.3. Proposed System

In this paper, we propose a three-stream 3D CNN with an SE block called an SE three-stream fusion network (SETFNet). We took three local regions, the eyes (including eyebrows), nose, and mouth, from the facial expression image sequence as inputs to the three-stream network. After fusions of the three streams, an SE block was added to the network to adaptively learn the weight of each feature channel.

To avoid over-fitting problems, a deep CNN requires large amounts of data for training. However, the available database for NIR expression is small in size. To train a CNN model on a small database, researchers use a medium-size CNN [39,40]. Therefore, the SETFNet in this paper was also a medium-size CNN with four convolutional layers.

The structure of the proposed SETFNet is shown in Figure 2. It is a three-stream 3D CNN consisting of three identical sub-networks. Each sub-network consists of four convolutional layers and has the same parameters. The number of convolution kernels for the four convolution layers, first through fourth, is 16, 32, 64, and 128, respectively. The kernel size of the first convolution layer is 3×3×8, and a large temporal stride here is used to eliminate some useless information. The kernel size of the other three convolution layers is 3×3×3. The three streams were fused and followed by an SE block to recalibrate the weight of each stream. The details of each subnetwork are shown in Table 1.

Table 1. Configuration of each stream.

Layers	Kernel Parameter Settings	Number of Kernels	Output Size
Date			32 × 36 × 64
Conv	3 × 3 × 8	16	9 × 18 × 32
Pool1	2 × 2 × 1	16	9 × 18 × 32
Conv2	3 × 3 × 3	32	9 × 9 × 16
Pool2	2 × 2 × 2	32	8 × 8 × 15
Conv3	3 × 3 × 3	64	8 × 8 × 15
Pool3	2 × 2 × 2	64	4 × 4 × 8
Conv4	3 × 3 × 3	128	2 × 4 × 8
Pool4	2 × 2 × 1	128	2 × 2 × 4

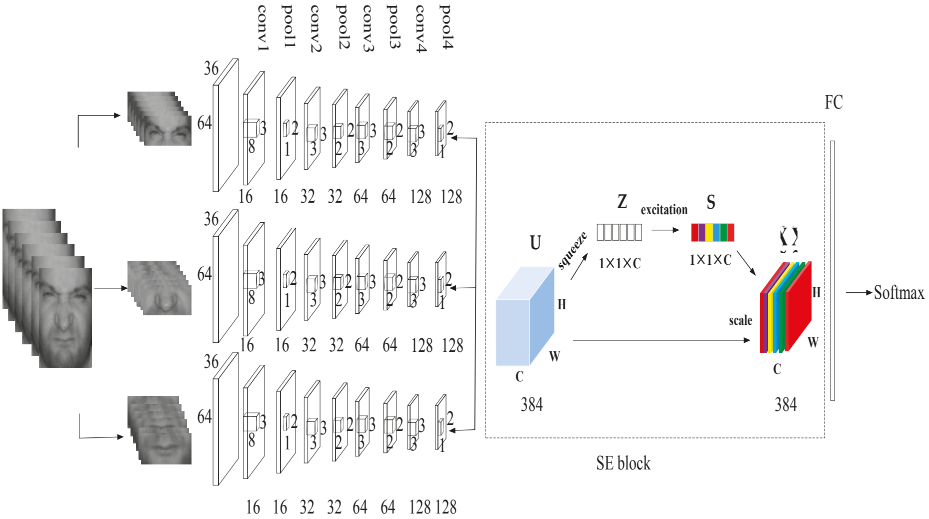


Figure 2. Overall structure of the proposed SE three-stream fusion network (SETFNet). The SE block is displayed in the dotted box.

Fusion Network

After extracting the features from the three regions (eyes, nose, and mouth), three stream features defined as T_1, T_2, and T_3 were obtained. The three stream features were then concatenated together to achieve better recognition by

$$T = T_1 \oplus T_2 \oplus T_{3,} \quad (4)$$

where T is the fused feature and ⊕ represents the concatenation operation. The concatenated features T were used as inputs to the next operation of the network.

3.4. Experiments

The proposed network was assessed on the Oulu-CASIA NIR facial expression database [18]. The network was implemented in the Caffe framework, which ran on a PC with a NVIDIA Geforce GTX 1080 graphical processing unit (GPU) (8 G). Training a model with the correct parameters is the key to achieving optimal performance, which has a direct impact on the experimental results. We trained the network from scratch using a batch size of 4, an initial learning rate of 10^{-3-3}, and a weight decay of 0.0005.

3.4.1. Database

Because the NIR facial expression database is not very common, the Oulu-CASIA NIR facial expression database is currently the only suitable one. It was collected in dark, weak, and normal light conditions, and consists of six kinds of facial expressions (anger, disgust, fear, happiness, sadness, and surprise) of 80 people between 23 and 58 years old, so each illumination condition has 480 image sequences. All expression sequences begin at the neutral emotion and end with the peak of the emotion. Each subject was asked to sit on a chair in the observation room in a way that they were in front of the camera. The distance between the face and camera was approximately 60 cm. Subjects made expressions according to the image sequences, while videos were captured by a USB 2.0 PC Camera (SN9C 201 & 202). Each clip was filmed by the camera at a frame rate of 25 fps. The image resolution was 320 × 240.

The aforementioned database has been used in many studies of facial expression recognition. It has been proved that the identification task under dark illumination conditions is the most difficult [18], because the facial image loses most of the texture features in dark light conditions. Therefore, we tested the proposed network on this most difficult sub-dataset (dark illumination condition).

We used the very popular method of tenfold cross-validation. All of the image sequences were divided into 10 groups. At each fold, nine groups were used to train the network and the rest were used for testing. During the entire experiment, there was no overlap between the training and testing sets.

3.4.2. Data Pre-Processing

In our experiment, a video sequence was pre-processed in the following three steps: (1) Frame-by-frame face detection; (2) locating eyes, nose, and mouth; and (3) cropping off the eyes, nose, and mouth areas. We found that step 2 had a significant effect on the performance of the network, so the choice of area to perform accurate spotting is crucial. To ensure that this was done accurately, the local areas were cropped based on the location of landmark points annotated by a robust landmark detector, discriminative response map fitting (DRMF) [41]. DRMF not only achieves good performance in landmark-detection methods [30], but also consumes very little computation time.

The cropping of these local areas was done by an automatic method. Since some of the cuts are inaccurate, manual cropping was used. Using the facial landmark points annotated earlier, the three regions were identified by using rectangular bounding boxes determined based on the eyes, nose, and mouth landmark points. We segmented the three local regions according to the following eleven points: E1 (x_1, y_1), E2 (x_2, y_2), E3 (x_3, y_3), E4 (x_4, y_4), E5 (x_5, y_5), N1 (x_6, y_6), N2 (x_7, y_7), M1 (x_8, y_8), M2 (x_9, y_9), M3 (x_{10}, y_{10}), and M4 (x_{11}, y_{11}) (shown in Figure 3). The center point of the rectangular bounding box of the eye region is L1 = E5 (x_5, y_5), and the length and width of the rectangle are $\frac{5}{3}|x_2 - x_1|$ and $\frac{4}{3}|y_4 - y_1|$, respectively. The center point of the rectangular bounding box of the nose region is L2 = $(x_5, \frac{y_7 - y_6}{2})$, and the length and width of the rectangle are $|y_7 - y_6|$ and $|x_3 - x_4|$, respectively. The center point of the rectangular bounding box of the mouth region is L3 = $(x_5, \frac{y_{11} - y_9}{2})$, and the length and width of the rectangle are $\frac{5}{3}|x_{10} - x_8|$ and $\frac{4}{3}$ $|y_{11} - y_9|$, respectively.

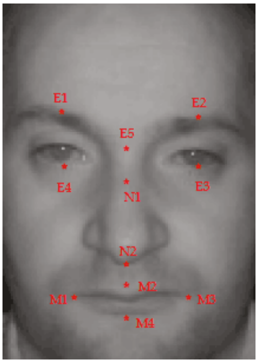


Figure 3. Positions of 11 points for segmenting three regions.

For the network input, each video sequence is normalized to 32 frames using the linear interpolation method [42]. Each frame of a global face (whole face) and local areas were resized to 88 × 108 and 36 × 64, respectively. To reduce the amount of calculation, all input images were converted to 8-bit grayscale.

4. Results and Discussion

4.1. Comparisons of Different Streams and Their Fusion

Table 2 shows the average results of tenfold cross-validation for each local region using a single sub-network (one stream) and a fused network. The feature information of the eye (including eyebrows), nose, and mouth regions is extracted by a single stream and the recognition rates are 35.37%, 42.76%, and 68.35%, respectively. The mouth region has the highest recognition rate, which may indicate that this part is the most expressive part in the database. The recognition rate of the eye region is the lowest among the three regions. This may be due to some of the participants wearing glasses. In the NIR face image, the NIR light reflected by the glasses removes the feature of the eyes, so that the frames with glasses have a great influence on recognition. At the same time, we can see that the performance of the recognition rate of the three-local-stream-fused networks (TFNets) reaches 78.68%, which is much higher than that of each single stream network (eye, 35.37%; nose, 42.76%; mouth, 68.35%). This indicates that our fusion is very effective in improving the recognition rate. After the network was fused, we added the SE block that automatically allocates weights to different streams. Since the SE block can make the entire network adaptively learn the weight of the feature channel, the SETFNet further improves the recognition rate, reaching a recognition rate of 80.34%.

Table 2. Comparison of different local and fused networks.

Architecture	Accuracy (%)	Time (s)
Eye	35.37	0.515
Nose	42.76	
Mouth	68.35	
TFNet	78.68	1.158
SETFNet	80.34	1.237
SETFNet + global	81.67	2.142

To investigate whether the SETFNet had extracted most of the expression features, we added one more stream to the SETFNet, which takes the frame of the global face as the input. Because each frame of the global face has larger spatial size than that of each local area, we added one more convolution pair to this added stream. The network structure is shown in Figure 4, with the fourth stream being the global face stream. When it is added to the SETFNet, the recognition rate becomes 81.67%. The SETFNet itself can achieve an 80.34% recognition rate. That is to say, after adding the entire face as input, the improvement of the recognition rate is still limited. This may indicate that the SETFNet has extracted most of the expression features.

Table 2 also shows the time consumption of various single sub-networks and fused networks. The time for a single sub-network to process an image sequence is 0.515 s, and the time for TFNet and SETFNet to process a sequence is 1.158 and 1.237 s, respectively. Considering the large improvement in recognition rate made by the TFNet and SETFNet, the increase of computation time is acceptable. However, when a global face stream is added to the SETFNet, the time for the network to process a sequence is 2.142 s. The slight increase in recognition rate (80.34% versus 81.67%) made by the global stream is at the expense of the processing time (1.237 s versus 2.142 s). However, all of the computation time may be within acceptable limits, since the input is 32 frames. Under the hardware settings used (NVIDIA Geforce GTX 1080 GPU (8G) for deep-learning acceleration), the SETFNet can process 32/1.237 = 25.87 frames every second. The frame rate of a normal imaging system is 25–30 fps, and 25.87 fps is within this range, which means that the SETFNet can give the recognition result just 1 s of lag in real-time imaging if the computation is performed in parallel with the imaging. With better hardware, the computation time can be further decreased to or to less than 1 s, which makes the processing a real-time process. Therefore, this network could be used in real applications.

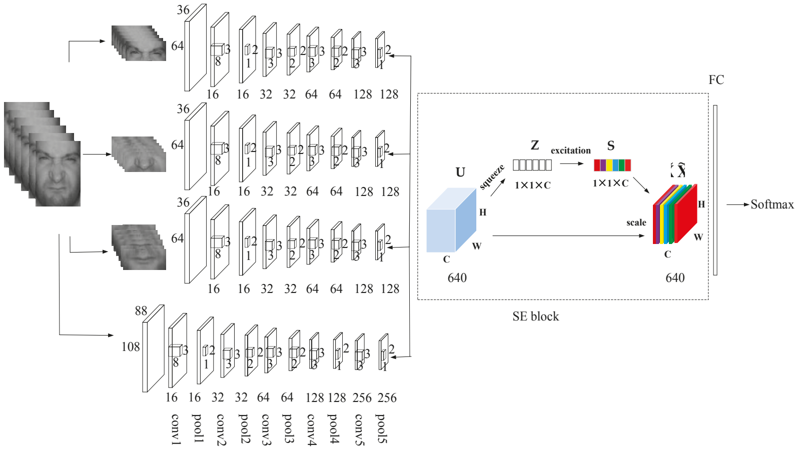


Figure 4. Structure of SETFNet plus global face stream.

The recognition rate of the eye region is the lowest among the three regions. One reason may be that the eyes have fewer features than the other parts; another reason could be that some of the subjects wear glasses. To verify the effect of glasses on the recognition rate, we input the eyes with and without glasses into the sub-network separately. The recognition results are shown in Table 3. It is seen that the recognition rate without glasses is better than that with glasses, which indicates that the glasses remove some features of the eyes. Since we divided the dataset into two parts, the recognition rates of wearing glasses and not wearing glasses are lower than that of the single sub-network with all data as the input.

Table 3. Comparison of recognition rate with and without glasses.

Category	Accuracy (%)
With glasses	30.13
Without glasses	31.45

4.2. Comparison of Embedded SE Block

The SE block was added to the network after the fusion so that the network could receive the information of the entire network and have a global receptive field. In the SE block, the reduction ratio *r* is an important parameter that can change the capacity and computational cost. We compared different reduction ratios *r* in our network model and the results are shown in the Table 4. When *r* = 16, the accuracy is the highest; therefore, *r* is set as 16.

Table 4. Comparison of different network reduction ratios.

Architecture		Accuracy (%)
SETFNet	r = 4	79.82
	r = 8	79.12
	r = 16	80.34
	r = 32	79.54
SETFNet + global	r = 4	80.57
	r = 8	81.25
	r = 16	81.67
	r = 32	80.38

4.3. Comparisons with Other Methods

Table 5 shows the different expression recognition rates of different methods on the Oulu-CASIA NIR facial expression database under dark-lighting conditions. For all of the methods, we used the tenfold cross-validation method to obtain an average recognition rate. The results of Deep Temporal Geometry Network (DTAGN), 3D CNN Deformable Facial Action Parts (DAP), and NIRExpNet were obtained from [37], and the result of LBP-TOP was obtained by implementing the algorithm using MatLab software (MathWorks, Natick, MA, USA). SETFNet and SETFNet + global were implemented by using Caffe. It is seen that LBP-TOP and 3D CNN DAP can achieve recognition rates of 69.32% and 72.12%, respectively, which are higher than that of DTAGN. NIRExpNet used the fusion information of local and global features, and therefore can achieve an even higher recognition rate than LBP-TOP and 3D CNN DAP. SETFNet uses only local information of three regions, but it can achieve a higher recognition rate (even higher than NIRExpNet, which uses local and global features). When a global face stream is added to SETFNet, it further improves the recognition rate to 81.67%. This indicates that the automatic allocation of the weight-of-features channel helps improve the recognition performance, which could be a promising method for NIR facial expression.

Table 5. Comparison of total recognition rates of different methods.

Method	Accuracy (%)
LBP-TOP [18]	69.32
DTAGN [43]	66.67
3D CNN DAP [27]	72.12
NIRExpNet [37]	78.42
SETFNet	80.34
SETFNet + global	81.67

4.4. Confusion Matrixes

To analyze the experimental results further, the confusion matrixes of SETFNet and SETFNet + global are shown in Tables 6 and 7, respectively. The labels on the left-hand side represent actual classes and those at the bottom represent the predicted classes; each percentage value in the matrix was calculated by dividing the number of a predicted class to the number of the corresponding actual class. After adding the global stream, the recognition rate of each expression is increased by 1–2%. It can be seen from Tables 6 and 7 that whether or not the global face stream is added, both happiness and surprise have high recognition rates, while fear and disgust have relatively low rates. The latter low recognition rates may be due to the slight movement of AUs for fear and disgust, which makes it more difficult to distinguish them from other expressions. Moreover, disgust is confused with anger, fear, and sadness, and fear is confused with anger, disgust, happiness, and surprise, perhaps because their appearance and movements are similar to each other.

Table 6. Confusion matrix of SETFNet. Labels on left-hand side represent actual classes; those on bottom represent predicted classes.

An	**77.64%**	**12.27%**	**1.25%**	**0**	**8.84%**	**0**
Di	15.06%	72.91%	9.53%	0	2.50%	0
Fe	7.45%	6.31%	68.53%	1.25%	0	16.46%
Ha	0	0	6.64%	93.36%	0	0
Sa	12.25%	3.52%	0	2.89%	81.34%	0
Su	0	0	8.46%	3.25%	0	88.29%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

Table 7. Confusion matrix of SETFNet + global. Labels on left-hand side represent actual classes; those on bottom represent predicted classes.

An	**78.43%**	**11.86%**	**0**	**0**	**9.71% ↑**	**0**
Di	13.38%	74.67%	7.87%	0	**4.08% ↑**	0
Fe	**9.54% ↑**	5.58%	71.08%	0	0	13.83%
Ha	0	0	5.74%	94.26%	0	**0**
Sa	**9.46%**	**8.25% ↑**	0	0	82.29%	0
Su	0	0	3.38%	**7.31% ↑**	0	89.31%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

SETFNet + global takes the entire face as input. The more input features there are, in general, should increase the true prediction values (values on the diagonal of the confusion matrix) and decrease the false prediction values (the zero value will be unchanged). It is seen from Table 6 that SETFNet + global does increase all true prediction values. However, more input does not always decrease the false prediction values. We can see from Table 7 that increased false prediction values do exist, which are indicated by up-pointing arrows. As the database is small in size, the prediction values could vary due to noise. To ensure that the located false prediction values are increased only as a result of more input features, we located their paired false prediction values as well. Each false prediction value pair appears in the same color in Table 7; for example, 9.54% (fear predicted as anger) and 0% (anger predicted as fear) in green. Only when both paired values are increased can the two expressions be considered as confused with each other more in SETFNet + global.

Under this criterion, we can see that sadness tends to be more recognized as disgust (8.25% versus 3.52%), or disgust tends to be more recognized as sadness (4.08% versus 2.50%), if SETFNet + global is used. The reason for this might be that, in sadness and disgust expression situations, lower cheek areas have an up-and-down movement pattern due to the movement of AU15 or AU10 [44]. When SETFNet + global takes these similar movement patterns as input, sadness will be recognized as disgust more.

Tables 8–11 show the confusion matrix of the comparison algorithms, with the labels on the left-hand side representing actual classes and those at the bottom representing the predicted classes. The confusion matrix of NIRExpNet (Table 8) was adopted from [37] directly. The other matrixes were obtained by implementing the algorithms with MatLab code on the database (tenfold cross-validation). Happiness and surprise again have higher recognition rates than the others in all algorithms. Fear has the lowest average recognition rate, and disgust has a similar average recognition rate to that of anger and sadness. This trend is in accord with what SETFNet reveals.

Table 8. Confusion matrixes of NIRExpNet.

An	**71.01%**	**14.43%**	**0**	**0**	**14.56%**	**0**
Di	20.56%	79.44%	0	0	0	0
Fe	0	8.00%	62.44%	0	0	29.56%
Ha	0	0	0	96.01%	0	3.99%
Sa	10.44%	0	14.44%	0	75.12%	0
Su	0	0	9.41%	4.04%	0	86.55%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

Table 9. Confusion matrixes of 3D CNN DAP.

An	**69.82%**	**16.23%**	**8.68%**	**0**	**5.27%**	**0**
Di	14.54%	73.41%	8.47%	0	3.58%	0
Fe	7.34%	7.46%	60.21%	8.32%	0	16.67%
Ha	0	0	8.58%	83.23%	0	8.19%
Sa	13.45%	9.93%	12.32%	0	64.30%	0
Su	**4.51%**	0	11.49%	2.45%	0	81.55%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

Table 10. Confusion matrixes of DTAGN.

An	**69.25%**	**15.28%**	**2.35%**	**3.30%**	**9.82%**	**0**
Di	18.72%	70.32%	10.96%	0	0	0
Fe	5.42%	3.13%	59.32%	5.62%	3.05%	23.46%
Ha	0	7.66%	12.57%	71.13%	0	8.64%
Sa	15.62%	0	14.52%	0	60.21%	9.65%
Su	0	0	13.46%	15.42%	0	71.12%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

Table 11. Confusion matrixes of LBP-TOP.

An	**63.45%**	**16.52%**	**7.66%**	**0**	**12.37%**	**0**
Di	15.33%	58.36%	10.67%	3.26%	12.36%	0
Fe	7.46%	6.89%	64.31%	0	3.89%	17.45%
Ha	0	11.68%	7.89%	75.86%	0	4.57%
Sa	10.62%	8.77%	10.43%	0	70.18%	0
Su	0	0	9.39%	6.85%	0	83.76%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

To further analyze the discrimination ability of different methods, we counted the number of zero false prediction values in each matrix. This number indicates that two corresponding expressions are perfectly recognized by the method. It is observed that NIRExpNet has 20 zero false prediction values, much more than other methods. 3D CNN DAP, DTAGN, and LBP-TOP have a similar number of zero false prediction values (approximately 12). These results indicate that NIRExpNet has the best performance in distinguishing one expression from others. This could be because NIRExpNet is designed specifically for the dataset. The features extracted by NIRExpNet are balanced so the possibility of confusing one expression with others is small.

Some zero false prediction values do not have zero paired values, e.g., the values in red in Table 9. 4.51% of the surprise expression was recognized as anger, but 0% anger was recognized as surprise using 3D CNN DAP. This could be due to the noise of the small dataset.

The F1 score and Matthews correlation coefficient (MCC) are calculated using the confusion matrixes, which are indexes considering accuracy and recall of the classification results and are fairer methods for assessing a classifier. The F1 score and MCC are summarized in Table 12. It is observed that SETFNet and SETFNet + global have the highest F1 and MCC, NIRExpNet has the second-highest values, and 3D CNN DAP the third highest. LBP-TOP and DTAGN have the lowest F1 and MCC. This indicates that SETFNet outperforms other methods in even more rigorous assessment. The order of the F1 and MCC performance of the methods is in accord with accuracy performance. This also indicates that the number of each sub-category is well balanced.

Table 12. Comparison of F1 score and MCC of different methods.

Method	F1 Score	MCC
LBP-TOP [18]	0.6712	0.6343
DTAGN [43]	0.6949	0.6077
3D CNN DAP [27]	0.7235	0.6702
NIRExpNet [37]	0.7828	0.7416
SETFNet	0.8034	0.7648
SETFNet + global	0.8164	0.7806

4.5. Potential Application and Improvement

SETFNet, which used three regions of the face as the input, can achieve higher recognition rates than NIRExpNet, which used the entire face as input, because an SE block can automatically allocate the weights to different streams. These results suggest that the automatic allocation of weights to different features will help improve the recognition rate. This idea of automatic allocation may have potential use in other recognition tasks. The SE block can always be added after a feature fusion step to allocate weights to different features to further improve the recognition rate.

SETFNet + global has a slightly higher recognition rate than SETFNet, but consumes much more calculation time. This indicates that a small part of the face could carry most of the expression information. For any other type of facial expression recognition task, we may only analyze the parts of face carrying expression information, which can save much calculation time and make recognition a real-time application.

The highest recognition rate on the Oulu-CASIA NIR facial expression database (dark condition) is 98.6%, achieved by Rivera et al. [45]. A number transitional graph method (DNG) was proposed in [45]. The confusion matrixes achieved by DNG method were summarized in Tables 13 and 14 (adopted from [45] directly), with the labels on the left-hand side representing actual classes and those at the bottom representing the predicted classes. Table 13 is the confusion matrix of DNG using 3D Sobel (DNG_S), and Table 14 is the confusion matrix of DNG using nine-plane mask (DNG_P). It is seen that the recognition rate of each expression class is more than 97% and similar to each other. This may indicate that the DNG has obtained good enough features to discriminate one expression from others. In terms of zero false prediction values, DNG_S has 21 zero false prediction values, and DNG_P has 23 zero false prediction values, which are less than all other methods. This indicates that the DNG method can achieve the most un-confused matrix. The F1 and MCC of DNG are higher than other methods, as well (DNG_S: F1 0.9859, MCC 0.9830; DNG_P: F1 0.9879, MCC 0.9856). This indicates that DNG outperforms other methods in more rigorous assessment.

Table 13. Confusion matrixes of DNG_S.

An	**98.75%**	**1.25%**	**0**	**0**	**0**	**0**
Di	2.53%	97.47%	0	0	0	0
Fe	0	0	97.81%	0.63%	1.25%	0.31%
Ha	0	0.63%	0	98.73%	0.63%	0
Sa	0	0	0	0.63	99.38%	0
Su	0	0	0.63%	0	0	99.38%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

Table 14. Confusion matrixes of DNG_p.

An	**100%**	**0**	**0**	**0**	**0**	**0**
Di	1.9%	96.2%	0	0	1.9%	0
Fe	0	0	99.38%	0	0.63%	0
Ha	0	0	0	98.73%	0.63%	0
Sa	0.63	0	0.63	0	98.75%	0
Su	0	0	0	0	0.63	99.38%
	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**

DNG consists of designed feature-extraction and feature-fusion methods, which make the extracted features robust in uneven illumination conditions. This could be the reason why DNG can achieve the best performance. According to the design of the DNG, two aspects could be considered in the future design of the SETFNet. Firstly, the uneven illumination conditions in the database could be taken into account when designing the network, such as using the features extracted from DNG as a stream to the network. Secondly, a more sophisticated fusion method could be used in future design, e.g., the concatenation operation used in this paper could be replaced by the fusion method in DNG.

However, a different form of DNG using hand-crafted features, SETFNet, proposed in this paper extracts features automatically. This design does not need the background knowledge of the data. Specifically, The feature extraction in this paper was finished by using a 3D CNN. Since the dataset used for training the CNN is small in size, the proposed network is not deep enough and may not extract high-level features. To further improve the recognition rate, transfer learning could be used, i.e., training a deeper CNN on a larger dataset and then fine-tuning the network on the NIR database.

5. Conclusions

In this paper, we proposed a three-stream 3D CNN architecture with an SE block called SETFNet that can automatically learn spatio-temporal features simultaneously. We only used three local regions of the face as input to the network. The advantages of using local information as input to the network were the removal of some information unrelated to recognition and a reduction of the amount of computation. To enable the network to adaptively learn the weight of each feature channel, an SE block was added to the network after the fusion of three single sub-networks. Experimental results show that SETFNet can achieve an average recognition rate of 80.34%; when a global face stream was added to SETFNet, the recognition rate was further increased to 81.67%, which is higher than some state-of-the-art methods.

Author Contributions: Data curation, L.Z., J.C., and Y.Y.; Formal analysis, Y.C.; Methodology, Z.Z.; Supervision, T.C.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 61301297, 61502398), and the Southwest University Undergraduate Science and Technology Innovation Fund (No.20600901).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Anderson, K.; McOwan, P.W. A real-time automated system for the recognition of human facial expressions. *IEEE Trans. Syst. Man Cybern. Part BCyben.* **2006**, *36*, 96–105. [CrossRef]
2. Ip, H.H.; Wong, S.W.; Chan, D.F.; Byrne, J.; Li, C.; Yuan, V.S.; Wong, J.Y. Enhance emotional and social adaptation skills for children with autism spectrum disorder: A virtual reality enabled approach. *Comput. Educ.* **2018**, *117*, 1–15. [CrossRef]
3. Tulyakov, S.; Slowe, T.; Zhang, Z. Facial expression biometrics using tracker displacement features. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, MN, USA, 18–23 June 2007; pp. 1–5.
4. Quintero, L.A.M.; Muñoz-Delgado, J.; Sánchez-Ferrer, J.C.; Fresán, A.; Brüne, M.; Arango de Montis, I. Facial emotion recognition and empathy in employees at a juvenile detention center. *Int. J. Offender Ther. Comp. Criminol.* **2018**, *62*, 2430–2446. [CrossRef] [PubMed]
5. Zeng, Z.; Pantic, M.; Roisman, G.I.; Huang, T.S. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 39–58. [CrossRef]
6. Bartlett, M.S.; Littlewort, G.; Fasel, I.; Movellan, J.R. Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction. In Proceedings of the 2003 IEEE Conference on Computer Vision and Pattern Recognition Workshop, Madison, WI, USA, 16–22 June 2003; Volume 5, p. 53.
7. Zhang, Z.; Wang, Y.; Zhang, Z. Face synthesis from low-resolution near-infrared to high-resolution visual light spectrum based on tensor analysis. *Neurocomputing* **2014**, *140*, 146–154. [CrossRef]
8. Wang, S.; Liu, Z.; Lv, S.; Lv, Y.; Wu, G.; Peng, P.; Wang, X. A natural visible and infrared facial expression database for expression recognition and emotion inference. *IEEE Trans. Multimedia.* **2010**, *12*, 682–691. [CrossRef]
9. Li, S.Z.; Chu, R.; Liao, S.; Zhang, L. Illumination invariant face recognition using near-infrared images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 627–639. [CrossRef]
10. Qiao, Y.; Lu, Y.; Feng, Y.S.; Li, F.; Ling, Y. A new method of NIR face recognition using kernel projection DCV and neural networks. In Proceedings of the 2013 Fifth International Symposium on Photoelectronic Detection and Imaging, Beijing, China, 25 June 2013; pp. 89071M1–89071M6.
11. Ekman, P.; Friesen, W.V. *Manual for the Facial Action Coding System*; Consulting Psychologists Press: Palo Alto, CA, USA, 1978.
12. Zeng, N.; Zhang, H.; Song, B.; Liu, W.; Li, Y.; Dobaie, A.M. Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing* **2018**, *273*, 643–649. [CrossRef]

13. Tsai, H.H.; Chang, Y.C. Facial expression recognition using a combination of multiple facial features and support vector machine. *Soft Comput.* **2018**, *22*, 4389–4405. [CrossRef]
14. Gu, W.; Xiang, C.; Venkatesh, Y.V.; Huang, D.; Lin, H. Facial expression recognition using radial encoding of local Gabor features and classifier synthesis. *Pattern Recognit.* **2012**, *45*, 80–91. [CrossRef]
15. Majumder, A.; Behera, L.; Subramanian, V.K. Automatic facial expression recognition system using deep network-based data fusion. *IEEE transactions on cybernetics.* **2018**, *48*, 103–114. [CrossRef] [PubMed]
16. Otberdout, N.; Kacem, A.; Daoudi, M.; Ballihi, L.; Berretti, S. Deep Covariance Descriptors for Facial Expression Recognition. *arXiv*, 2018; arXiv:1805.03869.
17. Zhao, G.; Pietikainen, M. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans. Pattern Anal. Mach.Intell.* **2007**, *29*, 915–928. [CrossRef]
18. Zhao, G.; Huang, X.; Taini, M.; Li, S.Z.; PietikäInen, M. Facial expression recognition from near-infrared videos. *Image Vis. Comput.* **2011**, *29*, 607–619. [CrossRef]
19. Ghimire, D.; Jeong, S.; Lee, J.; Park, S.H. Facial expression recognition based on local region specific features and support vector machines. *Multimed. Tools Appl.* **2017**, *76*, 7803–7821. [CrossRef]
20. Yan, W.J.; Wang, S.J.; Chen, Y.H.; Zhao, G.; Fu, X. Quantifying micro-expressions with constraint local model and local binary pattern. In Proceedings of the European Conference on Computer Vision workshop, Zurich, Switzerland, 6–12 September 2014; pp. 296–305.
21. Ringeval, F.; Schuller, B.; Valstar, M.; Jaiswal, S.; Marchi, E.; Lalanne, D.; Pantic, M. Av+ ec 2015: The first affect recognition challenge bridging across audio, video, and physiological data. In Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge. ACM, Brisbane, Australia, 26 October 2015; pp. 3–8.
22. Corneanu, C.A.; Simón, M.O.; Cohn, J.F.; Guerrero, S.E. Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications. *IEEE Trans. Pattern Anal. Mach.Intell.* **2016**, *38*, 1548–1568. [CrossRef] [PubMed]
23. Shan, C.; Gong, S.; McOwan, P.W. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vis. Comput.* **2009**, *27*, 803–816. [CrossRef]
24. Khan, S.A.; Hussain, A.; Usman, M. Reliable facial expression recognition for multi-scale images using weber local binary image based cosine transform features. *Multimed. Tools Appl.* **2018**, *77*, 1133–1165. [CrossRef]
25. Liu, M.; Shan, S.; Wang, R.; Chen, X. Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1749–1756.
26. Liu, P.; Zhou, J.T.; Tsang, I.W.H.; Meng, Z.; Han, S.; Tong, Y. Feature disentangling machine-a novel approach of feature selection and disentangling in facial expression analysis. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 151–166.
27. Liu, M.; Li, S.; Shan, S.; Wang, R.; Chen, X. Deeply learning deformable facial action parts model for dynamic expression analysis. In Proceedings of the 12th Asian Conference on Computer Vision, Singapore, 1–5 November 2014; pp. 143–157.
28. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231. [CrossRef] [PubMed]
29. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE conference on computer vision and pattern recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
30. Chen, J.; Chen, Z.; Chi, Z.; Fu, H. Facial expression recognition in video with multiple feature fusion. *IEEE Trans. Affect. Comput.* **2018**, *9*, 38–50. [CrossRef]
31. Fonnegra, R.D.; Díaz, G.M. Deep Learning Based Video Spatio-Temporal Modeling for Emotion Recognition. In Proceedings of the International Conference on Human-Computer Interaction, Las Vegas, NV, USA, 15–20 July 2018; pp. 397–408.
32. Yan, H. Collaborative discriminative multi-metric learning for facial expression recognition in video. *Pattern Recognit.* **2018**, *75*, 33–40. [CrossRef]
33. Zia, M.S.; Hussain, M.; Jaffar, M.A. A novel spontaneous facial expression recognition using dynamically weighted majority voting based ensemble classifier. *Multimed. Tools Appl.* **2018**, 1–31. [CrossRef]
34. Farokhi, S.; Sheikh, U.U.; Flusser, J. Near infrared face recognition using Zernike moments and Hermite kernels. *Inf. Sci.* **2015**, *316*, 234–245. [CrossRef]

35. Taini, M.; Zhao, G.; Li, S.Z. Facial expression recognition from near-infrared video sequences. In Proceedings of the 2008 IEEE International Conference on Pattern Recognition, Tampa, FL, USA, 18–21 December 2008; pp. 1–4.
36. Jeni, L.A.; Hideki, H.; Takashi, K. Robust Facial Expression Recognition Using Near Infrared Cameras. *JACIII* **2012**, *16*, 341–348. [CrossRef]
37. Wu, Z.; Chen, T.; Chen, Y.; Zhang, Z.; Liu, G. NIRExpNet: Three-Stream 3D Convolutional Neural Network for Near Infrared Facial Expression Recognition. *Appl. Sci.* **2017**, *7*, 1184. [CrossRef]
38. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
39. Peng, M.; Wang, C.; Chen, T.; Liu, G. Nirfacenet: A convolutional neural network for near-infrared face identification. *Information* **2016**, *7*, 61. [CrossRef]
40. Peng, M.; Wang, C.; Chen, T.; Liu, G.; Fu, X. Dual temporal scale convolutional neural network for micro-expression recognition. *Front. Psychol.* **2017**, *8*, 1745. [CrossRef] [PubMed]
41. Asthana, A.; Zafeiriou, S.; Cheng, S.; Pantic, M. Robust discriminative response map fitting with constrained local models. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 3444–3451.
42. Smolic, A.; Muller, K.; Dix, K.; Merkle, P.; Kauff, P.; Wiegand, T. Intermediate view interpolation based on multiview video plus depth for advanced 3D video systems. In Proceedings of the 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008; pp. 2448–2451. [CrossRef]
43. Jung, H.; Lee, S.; Yim, J.; Park, S.; Kim, J. Joint fine-tuning in deep neural networks for facial expression recognition. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2983–2991.
44. Ekman, P.; Friesen, W.; Hager, J. Facial Action Coding System The Manual. Available online: https://www.paulekman.com/product/facs-manual/ (accessed on 10 March 2019).
45. Rivera, A.R.; Chae, O. Spatiotemporal directional number transitional graph for dynamic texture recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *10*, 2146–2152. [CrossRef] [PubMed]

electronics

MDPI

Review

Socially Assistive Robots for Older Adults and People with Autism: An Overview

Ester Martinez-Martin *, Felix Escalona and Miguel Cazorla

Laboratory of Robotica and Vision Tridimensional (RoViT), University of Alicante, 03690 San Vicente del Raspeig (Alicante), Spain; felix.escalona@ua.es (F.E.); miguel.cazorla@ua.es (M.C.)

* Correspondence: ester@ua.es; Tel.: +34-965903400

Received: 16 January 2020; Accepted: 17 February 2020; Published: 21 February 2020

Abstract: Over one billion people in the world suffer from some form of disability. Nevertheless, according to the World Health Organization, people with disabilities are particularly vulnerable to deficiencies in services, such as health care, rehabilitation, support, and assistance. In this sense, recent technological developments can mitigate these deficiencies, offering less-expensive assistive systems to meet users' needs. This paper reviews and summarizes the research efforts toward the development of these kinds of systems, focusing on two social groups: older adults and children with autism.

Keywords: robotics; healthcare; disability; assistive technology; socially assistive robotics

1. Introduction

According to the World Health Organization (WHO) [1], one in seven people experience disability to some extent. However, only half can afford the required healthcare services [1]. This is especially critical when a person's quality of life diminishes and their independence is reduced. In this context, technological advances can play an important role, since they may enable people with disabilities to receive the healthcare necessary to lead a fulfilling life and be independent [2].

A review of the literature reveals the enormous variety of assistive technology currently available. Given the wide ranges of types and levels of deficiency, assistive technology can be classified depending on its complexity. Three concentric spheres of assistive technology can be defined with the user at their center. These are (from the inside to the outside): embodied assistive technology, assistive environments, and assistive robots.

Embodied assistive technology includes mobility devices [3,4] (e.g., wheelchairs, prostheses, exoskeletons, or artificial limbs); specialized aids (e.g., hearing [5], vision [6–8], cognition [9], or communication [10]); and specific hardware, software, and peripherals that assist people with disabilities with accessing information technologies (e.g., computers and mobile devices). Although these systems provide valued help, they usually offer just one functionality and lack much intelligence (intelligence being understood as the ability to receive feedback from the environment and adapt their behavior).

Going a step further, the environment can be adapted to the user's needs, with sensors and actuators, such as cameras or domotic systems, such that more functionalities are covered and more information about the user's health status can be gathered and processed, providing this technology with intelligence. Along those lines, we can find smart homes [11], virtual assistants [12–14] and ambient assisted living (AAL) settings [15–17]. Nevertheless, this kind of technology fails to support independent life when the user has chronic or degenerative limitations in motor and/or cognitive abilities.

As a solution, assistive robotics (AR) emerged. Its main goal is to fruitfully promote the well-being and independence of persons with disabilities. Robots may assist people in a wide range of tasks at home (especially in terms of activities for daily living), and so ongoing research includes household robots [18–20] and rehabilitation robots [21,22], among others. In the case of assistive robots, interdisciplinarity is required to achieve the final goal, integrating research areas such as artificial intelligence, human-robot interaction, and machine learning techniques, among others.

Thus, motivated by the current societal needs of the particular risk groups (i.e., children and older adults), this paper reviews and summarizes the promising and challenging research on assistive robotics aimed at helping older persons and children with autism to perform their daily tasks.

2. Socially Assistive Robots

One of the main difficulties in the acceptance of assistive technology is the way in which this technology is perceived. In this sense, the interaction between the robot and the user is a key issue. This social interaction led to the development of socially assistive robotics (SAR). According to Feil-Seifer and Mataric [23], SAR can be defined as the intersection of AR and socially interactive robotics (SIR), whose main task is interaction with human individuals.

Ideally, SAR should operate autonomously and not require the manipulation of a human operator. The interaction with the user must be intuitive and must not require extensive training. Additionally, the robots have to adapt their behaviors to the new routines and needs of the users, which is currently the most challenging task to be solved [24]. To meet this demand, artificial intelligence and machine learning algorithms must be developed and deployed in these systems, since the robots cannot be programmed in advance to react to every possible circumstance that might occur during interactions with the users.

As mentioned above, there exists a wide variety of applications depending on the needs to be covered and the demands of the target social group. Given that the SAR focuses on improving the user's life conditions, this study reviews the advances in two of the most vulnerable social groups:

- *Older adults;*
- *People with cognitive disorders.*

Section 3 reviews the latest advances in age-related health issues, while Section 4 analyzes the most significant research on children with autism in terms of diagnosis and therapy to train their communicative and social skills.

3. Older Adult Care

The aging population is one of today's major health concerns. This unprecedented situation urgently requires technological solutions to confront the constantly increasing demands of care services, which are currently overwhelmed. In this regard, the WHO identifies two key concepts in its *Global strategy and plan of action on aging and health* [25]:

- Healthy aging, understood as the process of developing and maintaining functional ability for older people's well-being;
- Functional ability, where technology is used to perform functions that might otherwise be difficult or impossible.

Healthy aging has become popular topic in recent decades. In this regard, SAR develops systems to improve older people's health through physical activity, which has a positive cognitive impact [26]. Some research attempts have consisted of companion robots that help users with assisted therapy and activity (see [27] for an overview). However, work is needed to promote for their acceptance among older people, as pointed out in [28], especially in terms of social interactions.

In addition, SAR for promoting physical exercise has been developed. This is, for instance, the case of the robotic *coach* proposed by Görer et al. [29]. It is essentially a technique based on a learning by imitation approach, which is used to learn the exercises from a human demonstrator. Then, the *reference* joint angles are used to evaluate the user's movements and to provide them with the necessary feedback to improve their performance. Note that two different platforms are used to achieve this goal. A NAO robot is used to describe the physical exercises, while an RGB-D camera captures the movements of the person. This can be problematic, since the correct position of the RBG-D device is essential to properly evaluate the user's performance. In addition, no sitting exercises are used because the skeleton data are insufficient to obtain the required results. Finally, the robot may confuse the user, given that it emulates the exercise as a demonstration and performs certain movements that are not to be carried out, such as head motions.

Another proposal is PHAROS [30,31], a socially assistive robot that monitors and evaluates the daily physical exercise done at the user's home (see Figure 1). For this, machine learning techniques (i.e., a convolutional neural network (CNN) together with a recurrent neural network (RNN)) are used to properly identify and evaluate the exercise performance. In addition, it integrates a recommender that generates the workout every day such that the person is working on what is necessary to stay healthy.

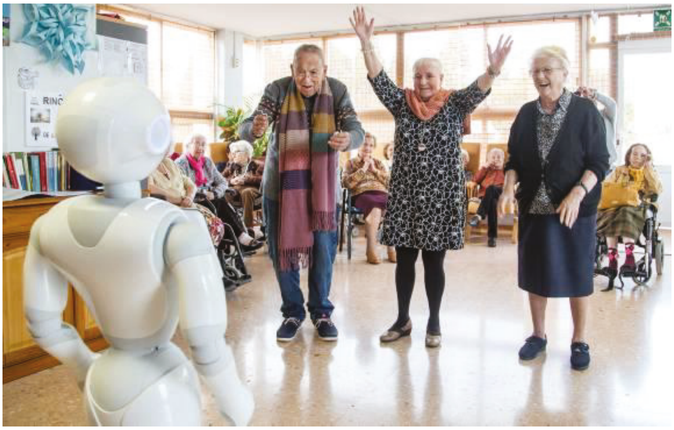


Figure 1. PHAROS robot in a pilot study at a residence of the elderly, Doña Rosa (Alicante).

Assisting functional ability requires more complex systems. In this sense, systems have been evolving over time, integrating an increasing number of functionalities. This is the case of the HOBBIT [32], a robot to help older people feel safe and continue to live in their own home. With this aim, the robot, illustrated in Figure 2, is able to autonomously navigate around the user's apartment, going anywhere they request, being able to pick up objects from the ground, bring a specific object, learn new objects to be found in the future, call in case of emergency, provide games for entertainment, and also remind the user to take their medication.

Analogously, the EU project RAMCIP [33] has developed a robotic assistant for older adults and those suffering from mild cognitive impairments (MCI) and dementia (see Figure 3). This robotic assistant also integrates several functionalities that promote physical and cognitive activity, such as detecting a fall (in which case a relative or external caregiver is informed), checking the cooker has been turned off after preparing a meal or the lights have been turned on when walking at night, picking up improperly left or fallen objects from the ground and moving them to safe storage, reminding users about their mediation, bringing the corresponding medicine and monitoring its taking, and facilitating social interactions with family and friends.

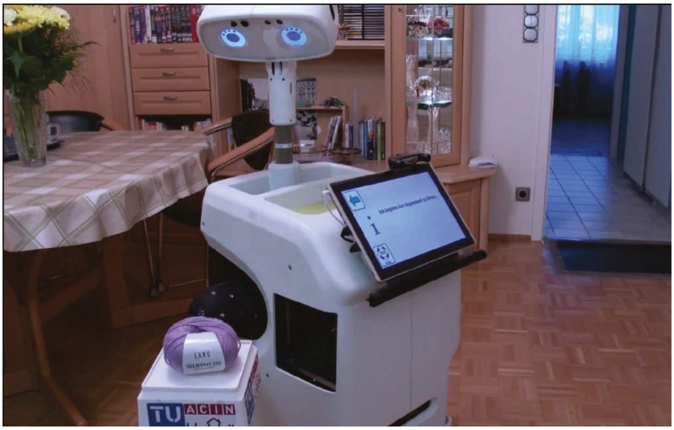


Figure 2. Hobbit robot in a pilot study at the Doña Rosa senior care home.

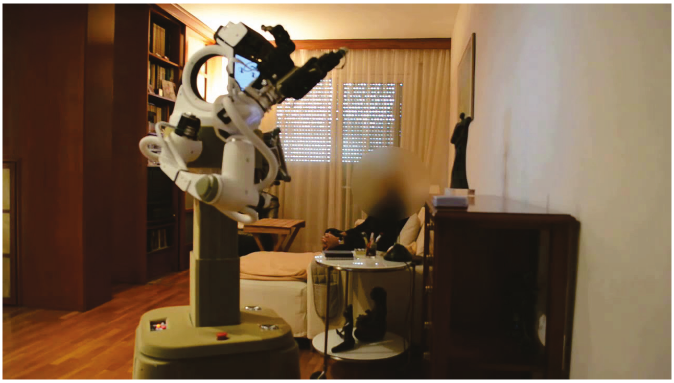


Figure 3. RAMCIP robot in a pilot study at a user's home.

Other solutions consider the possibility of integrating a robot platform into a smart home environment such that its functionalities may be augmented. An example is the robot activity support system (RAS) created by Washington State University [34] for adults with memory problems and other impairments to help them to live independently. Thus, the smart home has sensors in the walls to track the user's movement and feeds their data into the robot's neural network. This allows the robot to integrate activity detection technology to provide assistance when required. However, it is still at an early stage of development, and can only provide video instructions on how to do simple tasks, such as assisting a person through the steps of taking a dog for a walk or guiding them to an object. In addition, the need to install additional technology at home makes this option difficult and costly to implement.

Alternatively, other developments aim to assist people in nursing homes and healthcare facilities. In these kinds of systems, the key issue is the social component, with the aim being for the older adult user to perceive the robotic platform as a social companion rather than a machine to perform predefined tasks. This is the main focus of Rudy [35], an assistive robot created by INF Robotics in 2017. This robot offers telemedicine capabilities, such as remote patient monitoring (RPM), medication reminders, and medication dispensing (shown in Figure 4). In addition, it integrates a social component that, together with its friendly appearance, engages users. In fact, the social interactions are the most appreciated functionality of this system, since loneliness is a major issue among the aging. Nevertheless, it costs $5000, which is a significant amount which is not within all budgets.

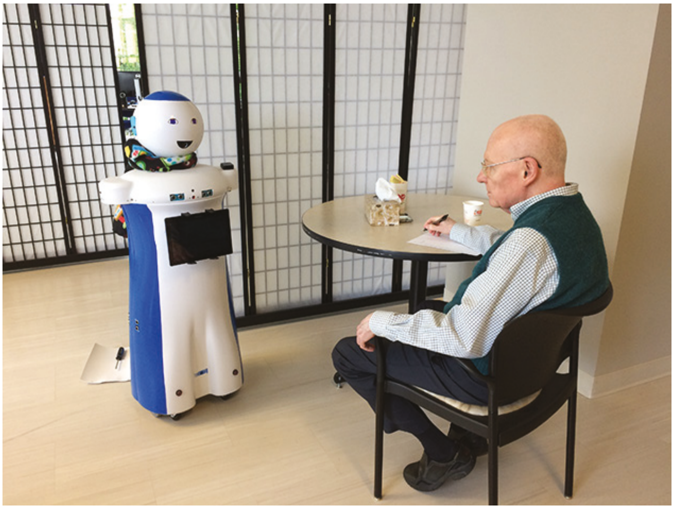


Figure 4. Rudy in a pilot study.

Along similar lines, Trinity College Dublin developed Stevie in 2017, which they improved in 2019 as Stevie II (Figure 5). The aim of this socially assistive IA robot was to augment the role of caregivers in long-term care environments, allowing them to concentrate mainly on person-centered tasks. Its functionalities range from medication reminders to keeping residents cognitively stimulated with quizzes and games. For this, enhanced expressive capabilities and a well-defined social component are used.

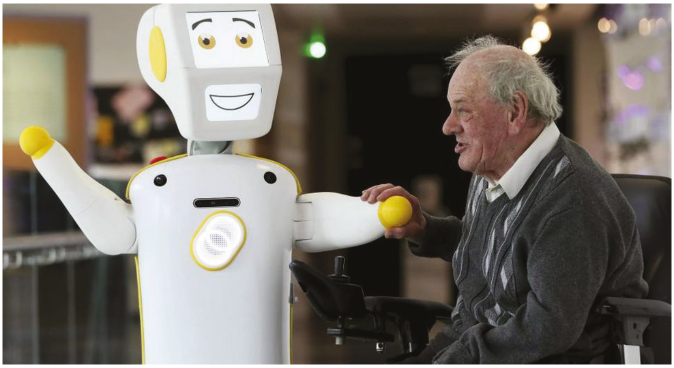


Figure 5. Stevie II in a pilot study.

4. Training Communication and Social Interaction in Children with Autism

In recent years, the use of SAR has become popular for the treatment and diagnosis of autism [36]. Indeed, the research in this field has presented an increase in user therapy acceptance and improvements in their social skills [37].

Applied behavior analysis (ABA) is one of the most extended therapies for the treatment of autism. It consists of improving specific behaviors, which are divided into simple and repetitive tasks that are presented sequentially and strategically while measuring and analyzing the patient's performance during the therapy [38].

The automation of some aspects of the therapy using technology with different devices and tools has been widely studied (videos, virtual and augmented reality, and robotics [39]). ABA therapies combined with SARs have exhibited substantial advantages and demonstrated their effectiveness in obtaining positive results in patients, such as high enthusiasm, increased attention, and increased social activity [40].

These results may be explained by the fact that children with autism feel more comfortable interacting with robots, because their behavior and reactions are more predictable [41]. Furthermore, the social skills of the patients could be gradually improved by increasing the complexity and unpredictability of the robot's behavior, making it more similar to actual human behavior [42].

These robotics systems can be used to manage therapy sessions, collect data and analyze the interactions with the patient, and generate information from this data in the form of reports and graphs. For this reason, they are a powerful tool for the therapist to check patient's progress and facilitate diagnosis.

The visual appeal of the robotics platform is a key factor to engaging the attention of children with autism. In general, these robots tend to use bright colors, rotating mechanical parts, striking shapes, and lights [43]. Additionally, some studies have reported that children with autism prefer to interact with robots with less humanoid characteristics [44]. However, some anthropomorphic robots have been succesfully used in research, especially in imitation and emotion recognition activities. Tables 1 and 2 present different SAR robots used in experiments. Following [45], there are several robot types depending on their location on the humanoid spectrum:

1. **Android**. They look like humans.
2. **Mascot**. They have humanoid forms but abstract or cartoonish appearances.
3. **Mechanical**. Humanoid forms with visibly mechanical parts.
4. **Animal**. Meant to look like pets.
5. **Non-Humanoid**. No resemblance to any living being.

Table 1. Robots used in autism therapies.

Robot	Appearance	Type	Description	Publications
Zeno R-50		Android	Child-sized robot (height = 0.64 m and weight = 6.5 kg) with a simplified expressive face. Its face has a motor that can be animated using software.	[46–48]
Nao		Android	Humanoid (height = 0.57 m and weight = 5 kg). Appearance of a human toddler. 11 DOF for its lower limbs and 14 DOF for its upper body.	[49–58]
Pepper		Android	Humanoid (height = 1.21 m and width = 0.48 m). It has almost the same articulations than a human, except for its mobile base and the impossibility of moving every finger independently. It has 4 microphones, two loudspeakers, two RGB cameras and a depth sensor (Asus Xtion). It has tactile sensors in the head and the back of its hands. It has a speech recognition engine that is able of identifying multiple variations in the human voice.	[59–62]
KASPAR		Android	Child-sized humanoid robot with minimal expressions. Can create body movements and gestures using its hands, arms, torso, head and show facial expressions.	[63–69]

Table 1. *Cont.*

Robot	Appearance	Type	Description	Publications
Keepon		Animal	Small creature-like robot (height = 12 cm). Simple, like a yellow snowman, and made of soft materials (silicone rubber).	[70–72]
Popchilla		Animal	Chinchilla-looking robot with movable arms, ears, mouth and eyes (teleoperated) with programmable speech output (Interbots). Provided with a iPad app.	[73]
PABI		Animal	Penguin-like small robot. 8 DOF in eyes, head, wings and opening beak. It carries a single board computer for autonomous operation and wireless communication for teleoperation. Speaker mounted behind its beak for communication. 2 independent video cameras in its eyes for tracking and monitoring. It carries a tablet as an interface with the onboard computer.	[74–76]
Pleo		Animal	Dinosaur-like robot. Developed to learn and repeat dances. 14 DOF, with movable legs, torso, neck, eyes, tail and mouth. Touch sensors in its whole body. Camera in its nose for object tracking and microphones. Capability to show emotions by making noises.	[77–80]
Robota		Android	Small robot (height = 45 cm and width = 14 cm) with the form of a young girl. 1 DOF of movement in its limbs (up and down), head rotation, 1 DOF for every arm, coordinated motion of the two eyes, individual blinking of the eyes and touch sensibility. Capabilities for vision tracking and machine learning.	[81–85]

Table 2. Robots used in autism therapies.

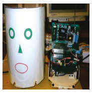
Robot	Appearance	Type	Description	Publications
i-Sobot		Android	Biped robot (height = 16.5 cm and weight = 350 g). 17 pieces of micro servo motors for walking and 180 different actions. 180 voice and sound commands. Remote controller or spoken commands.	[86–88]
Tito		Mascot	Robot (height = 17 cm) Coloured red, yellow and blue with washable clothes made of soft material. Wheels to move but with fake feet and legs to emulate human shape. Movable arms and head, lighting mouth for smiling. Wireless microphone-camera device inside one eye for tracking. Touch sensibility. Autonomous and teleoperated modes.	[89]
GIPY-1		Mechanical	Cylindrical mobile robot (diameter = 20 cm and height = 30 cm). Its face is the cladding of the robot: round eyes and nose triangle, with elliptical mouth. Can move forward, backward and turn on its own axis. Wireless controlled by a joystick.	[90,91]

Table 2. *Cont.*

Robot	Appearance	Type	Description	Publications
HOAP-3		Android	Humanoid robot (height = 60 cm and weight = 8.8 kg), commercialized by Fujitsu. 28 DOF for head, arms, legs and body movement. Inbuilt camera in its eyes for tracking and recognition. Microphones and speaker for audio recognition and speech. Expression LEDs to show emotions. Autonomous operation and teleoperated through WIFI.	[92]
Labo-1		Non-humanoid	Robotic mobile platform with form of a flat-topped buggy. 8 infrared sensors pointing in 4 directions for obstacle avoidance and a singLe positional heat sensor. Autonomous operation with an onboard computer and two buttons for behavior selection.	[93]
Ifbot		Mascot	Humanoid robot (height = 45 cm). 2 moving arms with 1 DOF and two wheels to move. 10 motors for facial expressions: eyes, eyelids and neck. 104 LEDS in its head and mouth to show emotions along with the facial expression.	[94]
Cosmobot		Mascot	Movable head, arm and mouth. Wheels to drive the robot in 4 directions. Pressure sensors and a built-in microphone for the interaction with the children. Expandable play station (Mission Control) for interaction, with external ports for joystick, wearable head and arm sensors. Teleoperated and controllable from a desktop computer software.	[95,96]
Ryan Companionbot		Android	Rear-projected humanoid. It shows 3D avatar models with speech and facial expressions. The animated face is projected into a face-shaped translucent mask. The 3D models are compatible with Maya design software.	[97–99]

Since the therapist's availability is limited, SARs must be developed with a certain level of autonomy in order to carry out the treatments. This autonomy is directly correlated with a SAR's level of intelligence in adapting to the environment and the patient's responses. This is where machine learning comes in, providing solutions to the problems these systems must address, such as eye-tracking, and face or automatic speech recognition.

Eye-tracking is the process of measuring the point of fixation of the gaze or the movement of an eye with respect to the head. It is used to measure a patient's attention to the robot. There exist commercial solutions for this purpose, but they are high cost or depend on special and invasive hardware (Tobii EyeX). However, there are many works focused on inferring the gazes of the users from images of their faces. Traditional techniques usually rely on shape-based methods, such as [100,101], observing geometries such as pupil centers and iris edges; and in appearance-based methods, such as [102,103], they directly use the images of the eyes for the prediction, with handmade features along with neural networks. In recent years, the focus has been on deep learning techniques to accomplish this task using standard and inexpensive camera devices. This is the case of [104], which uses a convolutional neural network to predict the gaze of the user from a color image of their face, previously trained with a large-scale dataset of faces and correlated gazes. More recent works such as [105] predict emotions and the patient's mood states from eye tracking data using recurrent neural networks.

The study of the patient's gaze is a crucial technique that helps with the diagnosis of autism and measures the effectiveness of the interaction between the robot and the user. In [106], the researchers carried out a study comparing the gaze attention of patients with autism when they interacted with humans and with robots . Similar to the previous example, in [107] the authors compare the gaze attention of people with autism while maintaining conversations with a human and a realistic android, which could serve as a diagnostic tool. In [84,85] the authors report the effects of repeated exposure to the humanoid robot Robota, which includes an increase in gaze attention and imitation.

Most of the experiments with these robots do not specify the kind of eye-tracking technique they use, or even whether they use external hardware, but recent works in this topic show that deep learning techniques outperform traditional ones without the need for invasive tools, so developments may move in this direction in order to ensure the best experience for users.

Face recognition has been one of the most widely studied research topics in computer vision since the beginnings of computer science, as it provides the recognition of subjects in a non-intrusive manner. The first step involves the detection and delimitation of the region of the image containing the face. Traditionally, detection has been conducted by searching for handcrafted features, like in [108], which uses cascade classifiers with different resolutions, trained with the Adaboost technique, based on Haar-like features. Subsequently, a vector of characteristics is extracted to describe the face, using global techniques like Eigenfaces [109] or Fisherfaces [110] based on Principal Component Analysis, or using local descriptors, like Local Binary Pattern Histograms [111], which codify the local structure of the image by comparing every pixel with its neighbourhood. However, traditional methods suffer when the conditions of the face are not ideal: recognition rates decrease with variations of the pose of the face and changes in the lighting conditions. Recent works have adopted end-to-end architectures based on deep learning that greatly outperform the traditional methods. Studies such as [112–115] use variations of convolutional neural network architectures trained with large-scale face datasets, obtained without pose restrictions, with good results on tests. Along with face recognition, recent studies like [116,117], classify the user's emotions by means of variants of convolutional neural networks, with promising results.

These characteristics are important for socially assistive robots in order to identify the patient and their mood and keep track of the history of the interaction. In [59], the researchers used face and emotion recognition to make a Pepper robot adapt a story to the mood of the children. In [118], the authors propose a technique for face recognition using a humanoid robot NAO to track the faces of the children with autism and measure their concentration during social interaction. In [61], the authors propose several activities through the interaction with a Pepper robot, receiving feedback by measuring the users' smiles.

Finally, *automatic speech recognition* is considered the most important bridge to enable human-machine communication. However, the technical difficulties of speech processing led to the keyboard and mouse becoming the most accurate interfaces for this purpose. Traditional methods in speech processing used statistical models, such as hidden Markov models [119,120], to process the wave signal and recognize the words pronounced and understand the sentences. However, these methods were very limited in vocabulary and the complexity of the sentences that human users could use and the recognition rates were far from perfect. Today, with the advent of GPUs, as in the previous sections, deep learning techniques are becoming the focus for researchers. End-to-end architectures, such as that proposed in [121–123], mainly based on a combination of convolutional neural networks, for extraction of features, and recurrent neural networks, for temporal information analysis, are taking the lead and obtaining interesting results.

In the case of social robotics, speech recognition is an important feature, as we need an intuitive, organic, and more natural method of communication than the old-fashioned peripherals. In [58], the researchers propose the use of the Nao robot to maintain conversations with children with autism and automatically extract crucial information on their interests to recommend them picture books. In [57], the authors propose a conversational therapy using a Nao robot that encourages the child to talk about their experiences and help them to recognize objects and imitate facial expressions. As a different approach, in [62], the authors use a Pepper robot to teach people with typical development to communicate with people with autism spectrum disorder.

All of these studies show that not only can patients with autism benefit from the advent of the SAR and artificial intelligence techniques, but therapists and family members also have more tools to help them with therapy and day-to-day living.

5. Conclusions

Socioeconomic changes and the lack of healthcare professionals to cover the unceasing demand of services and care have led to the need for technological solutions to mitigate this situation. In addition to intelligently interacting with the environment, the techniques developed must be successfully adopted by users. In this sense, neuroscientific evidence shows that users, especially children, tend to engage with robots better than traditional screens and their design must make the user feel comfortable and increase their well-being. As a consequence, the scientific response to these issues is assistive robotics, and more precisely, socially assistive robotics, which integrates a human-robot interaction in a social way.

This paper presents an overview of the state-of-the-art SAR solutions for helping and assisting older adults in their daily activities, such as activity scheduling and rehabilitation; and for helping children with autism spectrum disorders by means of diagnosis and social therapies. These solutions benefit from new advances in artificial intelligence, as these increase the autonomy levels of assistance robots, allowing them to adapt to unforeseen circumstances without the direct intervention of a human. Thus, the advent of SAR along with AI can help users with their day-to-day living, promoting their daily functioning, well-being, and independence.

Despite the active development in (social) assistive technology, there is still work to be done. Indeed, the current solutions do not provide ideal solutions to all needs of people with disabilities, but the results are highly promising.

Author Contributions: Conceptualization, E.M.M., F.E. and M.C.; methodology, E.M.M., F.E. and M.C.; software, E.M.M., F.E. and M.C.; validation, E.M.M., F.E. and M.C.; formal analysis,E.M.M., F.E. and M.C.; investigation, E.M.M., F.E. and M.C.; resources, E.M.M., F.E. and M.C.; writing—original draft preparation, E.M.M., F.E. and M.C.; writing—review and editing, E.M.M., F.E. and M.C.; visualization, E.M.M., F.E. and M.C.; supervision, E.M.M., F.E. and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Spanish Government TIN2016-76515-R grant for the COMBAHO project, supported with Feder funds. It has also been supported by Spanish grants for PhD studies ACIF/2017/243 and FPU16/00887.

Acknowledgments: Experiments were made possible by a generous hardware donation from NVIDIA.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. World Health Organization (WHO). Disability. 2019. Available online: https://www.who.int/disabilities/en/ (accessed on 19 February 2020).
2. Du Toit, R.; Keeffe, J.; Jackson, J.; Bell, D.; Minto, H.; Hoare, P. A Global Public Health Perspective. Facilitating Access to Assistive Technology. *Optom. Vis. Sci.* **2018**, *95*, 883–888, doi:10.1097/OPX.0000000000001272. [CrossRef]
3. Pant, P.; Gupta, V.; Khanna, A.; Saxena, N. Technology foresight study on assistive technology for locomotor disability. *Technol. Disabil.* **2018**, *29*, 163–171, doi:10.3233/TAD-170180. [CrossRef]
4. Tahsin, M.M.; Khan, R.; Gupta, A.K.S. Assistive technology for physically challenged or paralyzed person using voluntary tongue movement. In Proceedings of the 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), Dhaka, Bangladesh, 13–14 May 2016.
5. Abdallah, E.E.; Fayyoumi, E. Assistive Technology for Deaf People Based on Android Platform. *Procedia Comput. Sci.* **2016**, *94*, 295–301, doi:10.1016/j.procs.2016.08.044. [CrossRef]

6. Suresh, A.; Arora, C.; Laha, D.; Gaba, D.; Bhambri, S. Intelligent Smart Glass for Visually Impaired Using Deep Learning Machine Vision Techniques and Robot Operating System (ROS). In *Robot Intelligence Technology and Applications 5*; Springer: New York, NY, USA, 2018; pp. 99–112.
7. Phillips, M.; Proulx, M.J. Social Interaction Without Vision: An Assessment of Assistive Technology for the Visually Impaired. *Technol. Innov.* **2018**, *20*, 85–93, doi:10.21300/20.1-2.2018.85. [CrossRef]
8. Bhowmick, A.; Hazarika, S.M. An insight into assistive technology for the visually impaired and blind people: State-of-the-art and future trends. *J. Multimodal User Interfaces* **2017**, *11*, 149–172, doi:10.1007/s12193-016-0235-6. [CrossRef]
9. Palmqvist, L.; Danielsson, H. Parents act as intermediary users for their children when using assistive technology for cognition in everyday planning: Results from a parental survey. *Assist. Technol.* **2019**, 1–9, doi:10.1080/10400435.2018.1522523. [CrossRef]
10. Davydov, M.; Lozynska, O. Linguistic models of assistive computer technologies for cognition and communication. In Proceedings of the 2016 XIth International Scientific and Technical Conference Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 6–10 September 2016.
11. De Oliveira, G.A.A.; de Bettio, R.W.; Freire, A.P. Accessibility of the smart home for users with visual disabilities. In Proceedings of the 15th Brazilian Symposium on Human Factors in Computer Systems—IHC 16, São Paulo, Brazil, 4–7 October 2016.
12. Escalona, F.; Martinez-Martin, E.; Cruz, E.; Cazorla, M.; Gomez-Donoso, F. EVA: EVAluating at-home rehabilitation exercises using augmented reality and low-cost sensors. *Virtual Real.* **2019**, doi:10.1007/s10055-019-00419-4. [CrossRef]
13. Costa, A.; Novais, P.; Julian, V.; Nalepa, G.J. Cognitive assistants. *Int. J. Hum.-Comput. Stud.* **2018**, *117*, doi:10.1016/j.ijhcs.2018.05.008. [CrossRef]
14. Costa, A.; Novais, P.; Julian, V. A Survey of Cognitive Assistants. In *Intelligent Systems Reference Library*; Springer: New York, NY, USA, 2017; pp. 3–16.
15. Ruano, A.; Hernandez, A.; Ureña, J.; Ruano, M.; Garcia, J. NILM Techniques for Intelligent Home Energy Management and Ambient Assisted Living: A Review. *Energies* **2019**, *12*, 2203, doi:10.3390/en12112203. [CrossRef]
16. Costa, A.; Julián, V.; Novais, P. Advances and trends for the development of ambient-assisted living platforms. *Expert Syst.* **2016**, *34*, e12163, doi:10.1111/exsy.12163. [CrossRef]
17. Gomez-Donoso, F.; Escalona, F.; Rivas, F.M.; Cañas, J.M.; Cazorla, M. Enhancing the ambient assisted living capabilities with a mobile robot. *Comput. Intell. Neurosci.* **2019**, *2019*. [CrossRef] [PubMed]
18. ENRICHME: ENabling Robot and Assisted Living Environment for Independent Care and Health Monitoring of the Elderly. 2015. Available online: http://www.enrichme.eu/wordpress/ (accessed on 19 February 2020).
19. Paco Plus EU Project. 2010. Available online: http://www.paco-plus.org/ (accessed on 19 February 2020).
20. Cruz, E.; Escalona, F.; Bauer, Z.; Cazorla, M.; García-Rodríguez, J.; Martinez-Martin, E.; Rangel, J.C.; Gomez-Donoso, F. Geoffrey: An Automated Schedule System on a Social Robot for the Intellectually Challenged. *Comput. Intell. Neurosci.* **2018**, *2018*, doi:10.1155/2018/4350272. [CrossRef] [PubMed]
21. Luxton, D.D.; Riek, L.D. Artificial intelligence and robotics in rehabilitation. In *Handbook of Rehabilitation Psychology*, 3rd ed.; American Psychological Association: Washington, DC, USA, 2019; pp. 507–520.
22. Martinez-Martin, E.; Cazorla, M. Rehabilitation Technology: Assistance from Hospital to Home. *Comput. Intell. Neurosci.* **2019**, *2019*, doi:10.1155/2019/1431509. [CrossRef] [PubMed]
23. Feil-Seifer, D.; Mataric, M.J. Defining socially assistive robotics. In Proceedings of the 9th International Conference on Rehabilitation Robotics, ICORR 2005, Chicago, IL, USA, 28 June–1 July 2005; pp. 465–468.
24. Tapus, A.; Mataric, M.J. Socially Assistive Robots: The Link between Personality, Empathy, Physiological Signals, and Task Performance. In *AAAI Spring Symposium: Emotion, Personality, and Social Behavior*; AAAI Press: Boston, MA, USA, 2008; pp. 133–140.
25. World Health Organization (WHO). *Global Strategy and Action Plan on Ageing and Health*; Technical Report; World Health Organization Publications: Geneva, Switzerland, 2017.
26. Mura, G.; Carta, M.G. Physical Activity in Depressed Elderly. A Systematic Review. *Clin. Pract. Epidemiol. Ment. Health* **2013**, *9*, 125–135, doi:10.2174/1745017901309010125. [CrossRef]
27. Martinez-Martin, E.; del Pobil, A.P. Personal Robot Assistants for Elderly Care: An Overview. In *Intelligent Systems Reference Library*; Springer: New York, NY, USA, 2017; pp. 77–91.

28. Oh, S.; Oh, Y.H.; Ju, D.Y. Understanding the Preference of the Elderly for Companion Robot Design. In *Advances in Intelligent Systems and Computing*; Springer: New York, NY, USA, 2019; pp. 92–103.
29. Görer, B.; Salah, A.A.; Akın, H.L. An autonomous robotic exercise tutor for elderly people. *Auton. Robot.* **2016**, *41*, 657–678, doi:10.1007/s10514-016-9598-5. [CrossRef]
30. Martinez-Martin, E.; Costa, A.; Cazorla, M. PHAROS 2.0—A PHysical Assistant RObot System Improved. *Sensors* **2019**, *19*, 4531, doi:10.3390/s19204531. [CrossRef]
31. Costa, A.; Martinez-Martin, E.; Cazorla, M.; Julian, V. PHAROS—PHysical Assistant RObot System. *Sensors* **2018**, *18*, 2633, doi:10.3390/s18082633. [CrossRef]
32. EU Project. HOBBIT—The Mutual Care Robot. 2007–2013. Available online: http://hobbit.acin.tuwien.ac.at/ (accessed 19 February 2020).
33. EU Project. RAMCIP—Robotic Assistant for MCI Patients at Home. 2015–2020. Available online: https://ramcip-project.eu (accessed on 19 February 2020).
34. Wilson, G.; Pereyda, C.; Raghunath, N.; de la Cruz, G.; Goel, S.; Nesaei, S.; Minor, B.; Schmitter-Edgecombe, M.; Taylor, M.; Cook, D.J. Robot-enabled support of daily activities in smart home environments. In *Cognitive Systems Research*; Elsevier: Amsterdam, The Netherlands, 2019; Volume 54, pp. 258–272.
35. INF Robotics. Rudy. 2019. Available online: http://infrobotics.com/#rudy (accessed on 19 February 2020).
36. Dickstein-Fischer, L.A.; Crone-Todd, D.E.; Chapman, I.M.; Fathima, A.T.; Fischer, G.S. Socially assistive robots: Current status and future prospects for autism interventions. *Innov. Entrep. Health* **2018**, *5*, 15. [CrossRef]
37. Scassellati, B.; Admoni, H.; Matarić, M. Robots for use in autism research. *Annu. Rev. Biomed. Eng.* **2012**, *14*, 275–294. [CrossRef]
38. Kasari, C.; Lawton, K. New directions in behavioral treatment of autism spectrum disorders. *Curr. Opin. Neurol.* **2010**, *23*, 137. [CrossRef]
39. Goldsmith, T.R.; LeBlanc, L.A. Use of technology in interventions for children with autism. *J. Early Intensive Behav. Interv.* **2004**, *1*, 166. [CrossRef]
40. Begum, M.; Serna, R.W.; Yanco, H.A. Are robots ready to deliver autism interventions? A comprehensive review. *Int. J. Soc. Robot.* **2016**, *8*, 157–181. [CrossRef]
41. Scassellati, B. How social robots will help us to diagnose, treat, and understand autism. In *Robotics Research*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 552–563.
42. Dautenhahn, K.; Werry, I. Towards interactive robots in autism therapy: Background, motivation and challenges. *Pragmat. Cogn.* **2004**, *12*, 1–35. [CrossRef]
43. Cabibihan, J.J.; Javed, H.; Ang, M.; Aljunied, S.M. Why robots? A survey on the roles and benefits of social robots in the therapy of children with autism. *Int. J. Soc. Robot.* **2013**, *5*, 593–618. [CrossRef]
44. Robins, B.; Dautenhahn, K.; Dubowski, J. Does appearance matter in the interaction of children with autism with a humanoid robot? *Interact. Stud.* **2006**, *7*, 479–512. [CrossRef]
45. Ricks, D.J.; Colton, M.B. Trends and considerations in robot-assisted autism therapy. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 4354–4359.
46. Salvador, M.J.; Silver, S.; Mahoor, M.H. An emotion recognition comparative study of autistic and typically-developing children using the zeno robot. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 6128–6133.
47. Salvador, M.; Marsh, A.S.; Gutierrez, A.; Mahoor, M.H. Development of an ABA autism intervention delivered by a humanoid robot. In *International Conference on Social Robotics*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 551–560.
48. Silva, V.; Leite, P.; Soares, F.; Esteves, J.S.; Costa, S. Imitate Me!—Preliminary Tests on an Upper Members Gestures Recognition System. In *CONTROLO 2016*; Springer: Cham, Switzerland, 2017; pp. 373–383.
49. Geminiani, A.; Santos, L.; Casellato, C.; Farabbi, A.; Farella, N.; Santos-Victor, J.; Olivieri, I.; Pedrocchi, A. Design and validation of two embodied mirroring setups for interactive games with autistic children using the NAO humanoid robot. In Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23–27 July 2019; pp. 1641–1644.
50. Chevalier, P.; Isableu, B.; Martin, J.C.; Tapus, A. Individuals with autism: Analysis of the first interaction with nao robot based on their proprioceptive and kinematic profiles. In *Advances in Robot Design and Intelligent Control*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 225–233.

51. Lytridis, C.; Vrochidou, E.; Chatzistamatis, S.; Kaburlasos, V. Social engagement interaction games between children with Autism and humanoid robot NAO. In Proceedings of the 13th International Conference on Soft Computing Models in Industrial and Environmental Applications, San sebastian, Spain, 6–8 June 2018; pp. 562–570.
52. English, B.A.; Coates, A.; Howard, A. Recognition of Gestural Behaviors Expressed by Humanoid Robotic Platforms for Teaching Affect Recognition to Children with Autism-A Healthy Subjects Pilot Study. In Proceedings of the International Conference on Social Robotics, Tsukuba, Japan, 22–24 November 2017; pp. 567–576.
53. Qidwai, U.; Kashem, S.B.A.; Conor, O. Humanoid Robot as a Teacher's Assistant: Helping Children with Autism to Learn Social and Academic Skills. *J. Intell. Robot. Syst.* **2019**, 1–12. [CrossRef]
54. Shamsuddin, S.; Yussof, H.; Ismail, L.; Hanapiah, F.A.; Mohamed, S.; Piah, H.A.; Zahari, N.I. Initial response of autistic children in human-robot interaction therapy with humanoid robot NAO. In Proceedings of the 2012 IEEE 8th International Colloquium on Signal Processing and its Applications, Malacca, Malaysia, 23–25 March 2012; pp. 188–193.
55. Tapus, A.; Peca, A.; Aly, A.; Pop, C.; Jisa, L.; Pintea, S.; Rusu, A.S.; David, D.O. Children with autism social engagement in interaction with Nao, an imitative robot: A series of single case experiments. *Interact. Stud.* **2012**, *13*, 315–347. [CrossRef]
56. Petric, F.; Miklic, D.; Kovacic, Z. Robot-assisted autism spectrum disorder diagnostics using POMDPs. In Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, New York, NY, USA, 7 November 2017; pp. 369–370.
57. Mavadati, S.M.; Feng, H.; Salvador, M.; Silver, S.; Gutierrez, A.; Mahoor, M.H. Robot-based therapeutic protocol for training children with Autism. In Proceedings of the 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), New York, NY, USA, 26–31 August 2016; pp. 855–860.
58. Yang, X.; Shyu, M.L.; Yu, H.Q.; Sun, S.M.; Yin, N.S.; Chen, W. Integrating Image and Textual Information in Human–Robot Interactions for Children With Autism Spectrum Disorder. *IEEE Trans. Multimed.* **2018**, *21*, 746–759. [CrossRef]
59. Azuar, D.; Gallud, G.; Escalona, F.; Gomez-Donoso, F.; Cazorla, M. A Story-Telling Social Robot with Emotion Recognition Capabilities for the Intellectually Challenged. In Proceedings of the Iberian Robotics Conference, Porto, Portugal, 20–22 November 2019; pp. 599–609.
60. Burkhardt, F.; Saponja, M.; Sessner, J.; Weiss, B. How should Pepper sound-Preliminary investigations on robot vocalizations. *Stud. Zur Sprachkommun. Elektron. Sprachsignalverarbeitung* **2019**, *2019*, 103–110.
61. Nunez, E.; Matsuda, S.; Hirokawa, M.; Suzuki, K. Humanoid robot assisted training for facial expressions recognition based on affective feedback. In Proceedings of the International Conference on Social Robotics, Paris, France, 26–30 October 2015; pp. 492–501.
62. Yabuki, K.; Sumi, K. Learning Support System for Effectively Conversing with Individuals with Autism Using a Humanoid Robot. In Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 7–10 October 2018; pp. 4266–4270.
63. Wood, L.J.; Zaraki, A.; Walters, M.L.; Novanda, O.; Robins, B.; Dautenhahn, K. The iterative development of the humanoid robot kaspar: An assistive robot for children with autism. In Proceedings of the International Conference on Social Robotics, Tsukuba, Japan, 22–24 November 2017; pp. 53–63.
64. Wainer, J.; Robins, B.; Amirabdollahian, F.; Dautenhahn, K. Using the humanoid robot KASPAR to autonomously play triadic games and facilitate collaborative play among children with autism. *IEEE Trans. Auton. Ment. Dev.* **2014**, *6*, 183–199. [CrossRef]
65. Wainer, J.; Dautenhahn, K.; Robins, B.; Amirabdollahian, F. A pilot study with a novel setup for collaborative play of the humanoid robot KASPAR with children with autism. *Int. J. Soc. Robot.* **2014**, *6*, 45–65. [CrossRef]
66. Wood, L.J.; Zaraki, A.; Robins, B.; Dautenhahn, K. Developing kaspar: A humanoid robot for children with autism. *Int. J. Soc. Robot.* **2019**, 1–18. [CrossRef]
67. Huijnen, C.A.; Lexis, M.A.; de Witte, L.P. Matching robot KASPAR to autism spectrum disorder (ASD) therapy and educational goals. *Int. J. Soc. Robot.* **2016**, *8*, 445–455. [CrossRef]
68. Dautenhahn, K.; Nehaniv, C.L.; Walters, M.L.; Robins, B.; Kose-Bagci, H.; Mirza, N.A.; Blow, M. KASPAR—A minimally expressive humanoid robot for human–robot interaction research. *Appl. Bionics Biomech.* **2009**, *6*, 369–397. [CrossRef]

69. Robins, B.; Dautenhahn, K.; Dickerson, P. From isolation to communication: a case study evaluation of robot assisted play for children with autism with a minimally expressive humanoid robot. In Proceedings of the 2009 Second International Conferences on Advances in Computer-Human Interactions, Cancun, Mexico, 1–7 February 2009; pp. 205–211.
70. Kozima, H.; Michalowski, M.P.; Nakagawa, C. Keepon. *Int. J. Soc. Robot.* **2009**, *1*, 3–18. [CrossRef]
71. Kozima, H.; Nakagawa, C.; Yasuda, Y. Interactive robots for communication-care: A case-study in autism therapy. In Proceedings of the ROMAN 2005 IEEE International Workshop on Robot and Human Interactive Communication, Nashville, TN, USA, 13–15 August 2005; pp. 341–346.
72. Azmin, A.F.; Shamsuddin, S.; Yussof, H. HRI observation with My Keepon robot using Kansei Engineering approach. In Proceedings of the 2016 2nd IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), Ipoh, Malaysia, 25–27 September 2016.
73. Dunst, C.J.; Trivette, C.M.; Hamby, D.W.; Prior, J.; Derryberry, G. *Effects of Child-Robot Interactions on the Vocalization Production of Young Children with Disabilities. Social Robots. Research Reports, Number 4*; Orelena Hawks Puckett Institute: Asheville, NC, USA, 2013.
74. Woodyard, A.H.; Guleksen, E.P.; Lindsay, R.O. *PABI: Developing a New Robotic Platform for Autism Therapy*; Technical Report; Worcester Polytechnic Institute: Worcester, UK, 2015.
75. Brown, A.S. Face to Face with Autism. *Mech. Eng. Mag. Sel. Artic.* **2018**, *140*, 35–39. [CrossRef]
76. Dickstein-Fischer, L.A.; Pereira, R.H.; Gandomi, K.Y.; Fathima, A.T.; Fischer, G.S. Interactive tracking for robot-assisted autism therapy. In Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, Vienna, Austria, 6–9 March 2017; pp. 107–108.
77. Kim, E.S.; Berkovits, L.D.; Bernier, E.P.; Leyzberg, D.; Shic, F.; Paul, R.; Scassellati, B. Social robots as embedded reinforcers of social behavior in children with autism. *J. Autism Dev. Disord.* **2013**, *43*, 1038–1049. [CrossRef]
78. Kim, E.S.; Paul, R.; Shic, F.; Scassellati, B. Bridging the research gap: Making HRI useful to individuals with autism. *J. Hum.-Robot Interact.* **2012**, *1*, 26–54. [CrossRef]
79. Larriba, F.; Raya, C.; Angulo, C.; Albo-Canals, J.; Díaz, M.; Boldú, R. Externalising moods and psychological states in a cloud based system to enhance a pet-robot and child's interaction. *Biomed. Eng. Online* **2016**, *15*, 72. [CrossRef]
80. Curtis, A.; Shim, J.; Gargas, E.; Srinivasan, A.; Howard, A.M. Dance dance pleo: Developing a low-cost learning robotic dance therapy aid. In Proceedings of the 10th International Conference on Interaction Design and Children, Ann Arbor, MI, USA, 20–23 June 2011; pp. 149–152.
81. Dautenhahn, K.; Billard, A. Games children with autism can play with Robota, a humanoid robotic doll. In *Universal Access and Assistive Technology*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 179–190.
82. Billard, A.; Robins, B.; Dautenhahn, K.; Nadel, J. Building robota, a mini-humanoid robot for the rehabilitation of children with autism. *RESNA Assist. Technol. J.* **2006**, *19*, 37–49. [CrossRef]
83. Billard, A. Robota: Clever toy and educational tool. *Robot. Auton. Syst.* **2003**, *42*, 259–269. [CrossRef]
84. Robins, B.; Dautenhahn, K.; Te Boekhorst, R.; Billard, A. Effects of repeated exposure to a humanoid robot on children with autism. In *Designing a More Inclusive World*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 225–236.
85. Robins, B.; Dickerson, P.; Stribling, P.; Dautenhahn, K. Robot-mediated joint attention in children with autism: A case study in robot-human interaction. *Interact. Stud.* **2004**, *5*, 161–198. [CrossRef]
86. Watanabe, K.; Yoneda, Y. The world's smallest biped humanoid robot "i-Sobot". In Proceedings of the 2009 IEEE Workshop on Advanced Robotics and its Social Impacts, Tokyo, Japan, 23–25 November 2009; pp. 51–53.
87. Kaur, M.; Gifford, T.; Marsh, K.L.; Bhat, A. Effect of robot–child interactions on bilateral coordination skills of typically developing children and a child with autism spectrum disorder: A preliminary study. *J. Mot. Learn. Dev.* **2013**, *1*, 31–37. [CrossRef]
88. Srinivasan, S.M.; Lynch, K.A.; Bubela, D.J.; Gifford, T.D.; Bhat, A.N. Effect of interactions between a child and a robot on the imitation and praxis performance of typically devloping children and a child with autism: A preliminary study. *Percept. Mot. Ski.* **2013**, *116*, 885–904. [CrossRef] [PubMed]
89. Duquette, A.; Michaud, F.; Mercier, H. Exploring the use of a mobile robot as an imitation agent with children with low-functioning autism. *Auton. Robot.* **2008**, *24*, 147–157. [CrossRef]

90. Pradel, G.; Dansart, P.; Puret, A.; Barthélemy, C. Generating interactions in autistic spectrum disorders by means of a mobile robot. In Proceedings of the IECON 2010—36th Annual Conference on IEEE Industrial Electronics Society, Glendale, AZ, USA, 7–10 November 2010; pp. 1540–1545.
91. Giannopulu, I.; Pradel, G. Multimodal interactions in free game play of children with autism and a mobile toy robot. *NeuroRehabilitation* **2010**, *27*, 305–311. [CrossRef] [PubMed]
92. Ravindra, P.; De Silva, S.; Tadano, K.; Saito, A.; Lambacher, S.G.; Higashi, M. Therapeutic-assisted robot for children with autism. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 3561–3567.
93. Dautenhahn, K. Socially intelligent robots: dimensions of human–robot interaction. *Philos. Trans. R. Soc. Biol. Sci.* **2007**, *362*, 679–704. [CrossRef]
94. Lee, J.; Takehashi, H.; Nagai, C.; Obinata, G.; Stefanov, D. Which robot features can stimulate better responses from children with autism in robot-assisted therapy? *Int. J. Adv. Robot. Syst.* **2012**, *9*, 72. [CrossRef]
95. Lathan, C.; Boser, K.; Safos, C.; Frentz, C.; Powers, K. Using cosmo's learning system (CLS) with children with autism. In Proceedings of the International Conference on Technology-Based Learning with Disabilities, Edinburgh, UK, 15–17 August 2007; pp. 37–47.
96. Tzafestas, S. Sociorobot Field Studies. In *Sociorobot World*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 175–202.
97. Askari, F.; Feng, H.; Sweeny, T.D.; Mahoor, M.H. A Pilot Study on Facial Expression Recognition Ability of Autistic Children Using Ryan, A Rear-Projected Humanoid Robot. In Proceedings of the 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Nanjing and Tai'an, China , 27–31 August 2018; pp. 790–795.
98. Askari, F. *Studying Facial Expression Recognition and Imitation Ability of Children with Autism Spectrum Disorder in Interaction with a Social Robot*; Technical Report; University of Denver: Denver, CO, USA, 2018.
99. Mollahosseini, A.; Abdollahi, H.; Mahoor, M.H. Studying Effects of Incorporating Automated Affect Perception with Spoken Dialog in Social Robots. In Proceedings of the 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Nanjing and Tai'an, China, 27–31 August 2018; pp. 783–789.
100. Ishikawa, T. *Passive Driver Gaze Tracking with Active Appearance Models*; Technical Report; Robotics Institute, Carnegie Mellon University: Pittsburgh, PA, USA, 2004.
101. Valenti, R.; Sebe, N.; Gevers, T. Combining head pose and eye location information for gaze estimation. *IEEE Trans. Image Process.* **2011**, *21*, 802–815. [CrossRef]
102. Baluja, S.; Pomerleau, D. Non-intrusive gaze tracking using artificial neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 28 November–1 December 1994; pp. 753–760.
103. Sewell, W.; Komogortsev, O. Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. In Proceedings of the CHI'10 Extended Abstracts on Human Factors in Computing Systems, Denver, CO, USA, 10–11 December 2010; pp. 3739–3744.
104. Krafka, K.; Khosla, A.; Kellnhofer, P.; Kannan, H.; Bhandarkar, S.; Matusik, W.; Torralba, A. Eye Tracking for Everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
105. Sims, S.D.; Putnam, V.; Conati, C. Predicting Confusion from Eye-Tracking Data with Recurrent Neural Networks. *arXiv* **2019**, arXiv:1906.11211.
106. Damm, O.; Malchus, K.; Jaecks, P.; Krach, S.; Paulus, F.; Naber, M.; Jansen, A.; Kamp-Becker, I.; Einhaeuser-Treyer, W.; Stenneken, P.; et al. Different gaze behavior in human-robot interaction in Asperger's syndrome: An eye-tracking study. In Proceedings of the 2013 IEEE RO-MAN, Gyeongju, Korea, 26–29 August 2013; pp. 368–369.
107. Yoshikawa, Y.; Kumazaki, H.; Matsumoto, Y.; Miyao, M.; Kikuchi, M.; Ishiguro, H. Relaxing Gaze Aversion of Adolescents with Autism Spectrum Disorder in Consecutive Conversations with Human and Android Robot—A Preliminary Study. *Front. Psychiatry* **2019**, *10*, 370. [CrossRef]
108. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, 8–14 December 2001.
109. Turk, M.; Pentland, A. Eigenfaces for recognition. *J. Cogn. Neurosci.* **1991**, *3*, 71–86. [CrossRef] [PubMed]
110. Belhumeur, P.N.; Hespanha, J.P.; Kriegman, D.J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 711–720. [CrossRef]

111. Ahonen, T.; Hadid, A.; Pietikäinen, M. Face recognition with local binary patterns. In Proceedings of the European Conference on Computer Vision, Prague, Czech Republic, 11–14 May 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 469–481.
112. Sun, Y.; Liang, D.; Wang, X.; Tang, X. Deepid3: Face recognition with very deep neural networks. *arXiv* **2015**, arXiv:1502.00873.
113. Masi, I.; Wu, Y.; Hassner, T.; Natarajan, P. Deep face recognition: A survey. In Proceedings of the 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Parana, Brazil, 29 October–1 November 2018; pp. 471–478.
114. Yucel, M.K.; Bilge, Y.C.; Oguz, O.; Ikizler-Cinbis, N.; Duygulu, P.; Cinbis, R.G. Wildest faces: Face detection and recognition in violent settings. *arXiv* **2018**, arXiv:1805.07566.
115. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
116. Pramerdorfer, C.; Kampel, M. Facial expression recognition using convolutional neural networks: State of the art. *arXiv* **2016**, arXiv:1612.02903.
117. Kahou, S.E.; Bouthillier, X.; Lamblin, P.; Gulcehre, C.; Michalski, V.; Konda, K.; Jean, S.; Froumenty, P.; Dauphin, Y.; Boulanger-Lewandowski, N.; et al. Emonets: Multimodal deep learning approaches for emotion recognition in video. *J. Multimodal User Interfaces* **2016**, *10*, 99–111. [CrossRef]
118. Ismail, L.; Shamsuddin, S.; Yussof, H.; Hashim, H.; Bahari, S.; Jaafar, A.; Zahari, I. Face detection technique of Humanoid Robot NAO for application in robotic assistive therapy. In Proceedings of the 2011 IEEE International Conference on Control System, Computing and Engineering, Penang, Malaysia, 25–27 November 2011; pp. 517–521.
119. Juang, B.H.; Rabiner, L.R. Hidden Markov models for speech recognition. *Technometrics* **1991**, *33*, 251–272. [CrossRef]
120. Jelinek, F. *Statistical Methods for Speech Recognition*; MIT Press: Cambridge, MA, USA, 1997.
121. Amodei, D.; Ananthanarayanan, S.; Anubhai, R.; Bai, J.; Battenberg, E.; Case, C.; Casper, J.; Catanzaro, B.; Cheng, Q.; Chen, G.; et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 173–182.
122. Xiong, W.; Wu, L.; Alleva, F.; Droppo, J.; Huang, X.; Stolcke, A. The Microsoft 2017 conversational speech recognition system. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5934–5938.
123. Chan, W.; Jaitly, N.; Le, Q.; Vinyals, O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 4960–4964.

electronics

MDPI

Review

Identification of Daily Activites and Environments Based on the AdaBoost Method Using Mobile Device Data: A Systematic Review

José M. Ferreira [1,†], Ivan Miguel Pires [2,3,*,†], Gonçalo Marques [2,†], Nuno M. Garcia [2,†], Eftim Zdravevski [4,†], Petre Lameski [4,†], Francisco Flórez-Revuelta [5,†] and Susanna Spinsante [6,†]

[1] Computer Science Department, Universidade da Beira Interior, 6200-001 Covilhã, Portugal; jose.ferreira@ubi.pt

[2] Instituto de Telecomunicações, Universidade da Beira Interior, 6200-001 Covilhã, Portugal; goncalosantosmarques@gmail.com (G.M.); ngarcia@di.ubi.pt (N.M.G.)

[3] Computer Science Department, Polytechnic Institute of Viseu, 3504-510 Viseu, Portugal

[4] Faculty of Computer Science and Engineering, University Ss Cyril and Methodius, 1000 Skopje, Macedonia; eftim.zdravevski@finki.ukim.mk (E.Z.); petre.lameski@finki.ukim.mk (P.L.)

[5] Department of Computing Technology, University of Alicante, P.O. Box 99, E-03080 Alicante, Spain; francisco.florez@ua.es

[6] Department of Information Engineering, Università Politecnica delle Marche, 60131 Ancona, Italy; s.spinsante@staff.univpm.it

* Correspondence: impires@it.ubi.pt; Tel.: +351-966-379-785

† These authors contributed equally to this work.

Received: 9 December 2019; Accepted: 13 January 2020; Published: 20 January 2020

Abstract: Using the AdaBoost method may increase the accuracy and reliability of a framework for daily activities and environment recognition. Mobile devices have several types of sensors, including motion, magnetic, and location sensors, that allow accurate identification of daily activities and environment. This paper focuses on the review of the studies that use the AdaBoost method with the sensors available in mobile devices. This research identified the research works written in English about the recognition of daily activities and environment recognition using the AdaBoost method with the data obtained from the sensors available in mobile devices that were published between 2012 and 2018. Thus, 13 studies were selected and analysed from 151 identified records in the searched databases. The results proved the reliability of the method for daily activities and environment recognition, highlighting the use of several features, including the mean, standard deviation, pitch, roll, azimuth, and median absolute deviation of the signal of motion sensors, and the mean of the signal of magnetic sensors. When reported, the analysed studies presented an accuracy higher than 80% in recognition of daily activities and environments with the Adaboost method.

Keywords: daily activities recognition; ensemble learning; ensemble classifiers; environments; mobile devices; sensors; systematic review

1. Introduction

AdaBoost is one of the first boosting algorithms developed by Yoav Freund and Robert Schapire that was adapted for practical application in many solving tasks. AdaBoost is a method that uses ensemble learning techniques to combine multiple weak classifiers into a single strong classifier. It is combined with other artificial intelligence methods to increase the accuracy of the recognition [1]. Thus, weak learners, including decision tree and decision boosting, are commonly used with the AdaBoost method. In comparison with other machine learning methods, the AdaBoost method is less susceptible to overfitting.

One of the strategies adopted by the different implementation of Adaboost consists in combination with other methods to reduce the errors obtained [2,3]. The primary purpose of ensemble learning techniques is to improve the results by combining the results of different methods [2,3]. These techniques consist of the combination of several machine learning techniques with a single purpose and model to improve the prediction results [4–6]. It can be divided into two groups, sequential ensemble methods and parallel ensemble methods, where our focus is the sequential ensemble methods, because the implementation of Adaboost consists in the application of a base learner that is generated sequentially [7].

In the last years, several studies have been developed with a focus on the recognition of daily activities using the sensors available in the commonly used mobile devices. These studies conclude that it is possible to accurately detect the daily activities and environments with motion, magnetic, location and acoustic sensors embedded on mobile devices, reporting reliable results available in the literature with different machine learning methods [8–23].

To date, and due to the increasing power processing capabilities of the different mobile devices, the Adaboost method is one of the most used methods, and it reports reliable results [24–32]. The motivation of this systematic review is to evaluate the reliability of the Adaboost method for daily activities and environment recognition using the sensors available in mobile devices for further implementation of a framework [33–42].

Generally, the raw readings of one-dimensional (e.g., blood pressure sensor, thermometer, etc.) or multi-dimensional signals (e.g., accelerometer or gyroscope) can be directly processed by AdaBoost, and other classification and regression algorithms in general. To do that, all sensory readings in a specific time window represent different inputs. For example, if a thermometer reads data with 1 Hz frequency, and the window is 60 s, there will be 60 inputs to AdaBoost. Similarly, a three-dimensional gyroscope would present 180 inputs. Many deep learning methods accept the input data in this format. Be that as it may. Usually, many algorithms benefit from a feature engineering step [43], which significantly improves the accuracy or simplifies the complexity of the models [23,44].

Due to the complex nature of the sensory data collected using the sensors available in mobile devices, the overfitting problem is impacts many machine learning algorithms, including multilayer perceptron neural networks (MLP), deep neural networks (DNN) and feedforward neural networks (FNN) [33–42]. Methods for parameter tuning such as grid search [45] and systematic feature selection [23] are usually applied to mitigate this problem.

Previous studies [33–42] shown that the proposed framework includes the correct modules for the reliable recognition of daily activities and environments. However, the results can be improved with other methods, including ensemble learning methods.

This paper reviews the different studies available in the literature related to the implementation of the AdaBoost method for daily activities recognition. This review is included in the research and development of a framework associated with the identification of daily activities and environments using the sensors available in mobile devices, where the AdaBoost method can increase the accuracy compared to other implementations. The motivation of this paper is to improve the accuracy reported in previous studies for the recognition. This review intends to explore the use of the Adaboost method to verify if it reports better results than MLP, FNN, and DNN methods for the identification of daily activities.

The main contribution of this review is the presentation of a base of study for the readers who deal with the recognition of daily activities and environments using sensors available in mobile devices providing an in-depth survey of several research projects which implement Adaboost method.

This review shows that the features that reported better results are mean, standard deviation, pitch, roll, azimuth and median absolute deviation of the signal of motion sensors, and the mean of the signal of magnetic sensors. According to the results, the Adaboost method provides huge accuracy for the recognition of daily activities and environments.

The following sections are organized as follows: Section 2 presents the methodology of the review. The results obtained are presented in Section 3. Section 4 presents the discussion on the results. Finally, the conclusions are presented in Section 5.

2. Methodology

2.1. Research Questions

In this way, the leading questions of this review are: (RQ1) What is AdaBoost? (RQ2) How to detect daily activities with AdaBoost? (RQ3) How to identify daily activities with AdaBoost using mobile devices?

2.2. Inclusion Criteria

Studies assessing the recognition of daily living using AdaBoost method were included in this review according to the following criteria: (1) Detect daily activities using sensors; (2) implementing AdaBoost method for the automatic recognition of daily activities, presenting the information about the activities and environments recognized; (3) make use of mobile devices; (4) presents the accuracies obtained with AdaBoost method; (5) published between 2010 and 2019; (6) were available in open-access libraries; and (7) written in English.

2.3. Search Strategy

The authors of this review searched for studies according to the inclusion criteria in the following electronic databases: IEEE Xplore and Science Direct. Every study was independently evaluated by eight reviewers (JF, IMP, GM, NMG, EZ, PL, FFR, and SS), and all parties evaluated its suitability. The studies were examined to identify the characteristics of AdaBoost and its relevance for the implementation in recognition of daily activities and environments using mobile devices.

2.4. Extraction of Study Characteristics

The following data were extracted from the studies and tabulated (see Tables 1 and 2): Year of publishing, the population was taken into account, purpose, equipment used, and outcomes of each publication. All cited studies in Tables 1 and 2 informed that the experiments were performed in laboratory settings. The verification of the availability of the raw data was performed.

Table 1. Study summaries.

Authors	Year	Outcomes
Kelarev et al. [46]	2012	A cardiovascular autonomic neuropathy identification algorithm that uses mobile devices is proposed. The dataset has been created using health records collected in a university research project named Diabetes Complications Screening Research Initiative. The main contribution of the paper is the recommendation of the AdaBoost and Bagging based on the J48 decision.
Xu et al. [47]	2014	The paper presents an accurate method for context detection, which uses multiple sensors and machine learning. The context information is restrictively used to select activities that require classification, increasing the accuracy and decreasing the complexity of the process. Fourteen subjects each carried a tablet, and four 9-DOF sensors were located on wrists, ankle, knee, and mid-waist. Each volunteer allocated thirty minutes in every context and did each required activity from two to five minutes. The dataset was then divided into two parts, 30% of the data for training and 70% rest for testing. The combined results of the three classifiers were able to achieve higher accuracy for all contexts.
Wisniweski et al. [48]	2014	The paper presents an automatic recognition method of asthmatic wheezing through the analysis of a breathing sound dataset. One hundred thirty s130 records for natural and wheezy breathing using 1024 samples each were used for the study. The overall recognition was 93%.
Zhou et al. [49]	2015	The authors propose the HATS, which provides both entry-point and post-log-in mobile user authentication. The proposed method integrates several authentication methods like password, keystroke, gesture, and touch dynamics features to explore the vulnerabilities of specific approaches to specific security attacks. The participants were required to go through several training sessions to be introduced to the usage of two different keyboards. Twelve volunteers (for men, eight women) carried the study.
Masri et al. [50]	2015	The study proposes active authentication applying scrolling behaviors for biometrics and evaluates diverse classification and clustering approaches that support those characteristics. The experiment counted with 84 participants and 54 documents. The most accurate method was achieved adopting k-means clustering among two techniques applied to validate users, with a success rate of 83.5%.
Xu et al. [51]	2016	The authors propose an online learning approach for activity recognition based on data collected using inertial sensors. The data was gathered from fourteen volunteers. Every volunteer performs thirty minutes in the respective context and carried each required activity for two to five minutes. This algorithm outperformed the benchmark algorithms by 30–40%.
Tang et al. [52]	2016	The paper shows an assessment of ten representative classifiers applied in two datasets. The dataset contains accelerometer time-series data from 22 volunteers. This study concluded that K-Nearest Neighbors is the most suitable classifier.
Yanyun et al. [53]	2017	The paper presents a method based on Convolutional Neural Networks approach to provide automatic extraction of features for transportation mode classification. There were used a total of 169 features, and the dataset has more than 200 h of transportation data collected from thirty volunteers on diverse transportation modes (bus, car, metro, train). The recognition accuracy was: 96.6% for the bus, 99.6% for the car, 99.0% for the metro, and 98.9% for the train. Giving an average accuracy of 98.6%.

Table 1. *Cont.*

Authors	Year	Outcomes
Li et al. [54]	2017	The authors propose an indoor and outdoor recognition method, which is divided into two parts: The machine learning-based Indoor, Outdoor, and Semi-open areas recognition algorithm and the lightweight WiFi sub-detector. The absolute values and the relative measurements of WiFi received signal strength are calculated to identify if the user environment is a semi-open area, indoor or outdoor. The proposed method presents 85% of accuracy for the lightweight WiFi-based technique and 96% of accuracy using the aggregated IOS-detector.
Yanjun et al. [55]	2017	The article proposes a Bayesian algorithm for traffic pattern recognition. The used dataset consists of 400 h from eight individuals. An accelerometer, a barometer, a geomagnetic, a gyroscope, and base station were the five used sensors. The AdaBoost classification method was also implemented to get better results. The proposed method presents an accuracy rate from 83.3% to 91.5%.
Vafeiadis et al. [56]	2017	The paper presents a machine learning approach for occupancy detection. The water and energy consumption data collected using smart meters are used as features for occupancy detection in a domestic environment. Under their boosting versions, Random Forest and Decision Tree classifiers present more accuracy when associated with the other classifiers. The authors obtain an overall accuracy of 83.37% and 82.79%, respectively.
Subasi et al. [57]	2018	This study proposes the use of AdaBoost based classifier for human activity recognition using data collected from sensors located on the body. The study is based on nine inertial sensors collected by seventeen volunteers who perform 33 fitness exercises. The results present 99.98% of success rate.
Yuan et al. [58]	2018	The authors present an indoor localization algorithm based on 'Twi-AdaBoost'. The proposed method uses several sensors, such as gyroscope, magnetometer, and accelerometer. The tests used 6304 samples collected from both smartphone and smartwatch devices. The AdaBoost method outperforms the other approaches tested in every metrics.

Table 2. Critical analysis of reviewed studies.

Authors	Population	Purpose of Study	Device Type	Public Dataset	Pros	Cons
Kelarev et al. [46]	Not Mentioned	Cardiovascular autonomic neuropathy identification on mobile devices	Mobile Devices	Yes	Evaluated multiple ensemble strategies; Novel ensemble of AdaBoost and Bagging based on J48	Dataset is scarcely described; Processing pipeline not illustrated; Not evaluated on different test subjects
Xu et al. [47]	Fourteen individuals use an Android tablet, and four 9-DOF (Degrees of freedom) sensors were located on the wrist, knee, ankle, and mid-waist	An end-to-end system is proposed to enable large-scale supervision and classification of physical movements	Mobile Devices	Yes	End-to-end system that integrates context into activity classification; 13 activities of daily living; 8 environments; Battery life considerations	The authors should use models able to learn the activities associated with contexts in conjunction with scenarios; The authors should test the robustness of the system and improve privacy, security, and user friendliness; Not evaluated on different test subjects

Table 2. *Cont.*

Authors	Population	Purpose of Study	Device Type	Public Dataset	Pros	Cons
Wisniweski et al. [48]	Population not mentioned, 130 records for natural and wheezy breathing using 1024 samples each	An automatic and highly efficient method is proposed for asthmatic wheezing recognition in breathing sounds	Mobile Devices	Yes	It detects asthmatic wheezes; The implemented method is not computationally complex; it is capable of monitoring asthma using mobile devices	The authors should test other machine learning methods; Not evaluated on different test subjects; Dataset is scarcely described
Zhou et al. [49]	Twelve individuals (four men and eight women)	A harmonized user authentication method based on ThumbStrokes dynamics (HATS) for smartphone touch screen	Smartphones	No	It supports entry-point and post-login mobile user authentication; It explores previous studies to implement a better solution; It improves the security and authentication of mobile authentication systems	The study needs to be tested using a larger sample size; The authors does not perform feature reduction, feature selection and data transformation; Not evaluated with different test subjects and devices
Masri et al. [50]	84 individuals participated in the experiment and 54 documents were available for reading	Discusses active authentication methods which utilise scrolling behaviors for bio-metrics and evaluates diverse classification, and clustering approaches that lead to those characteristics	Mobile Devices	Yes	The data used is composed by data acquired from the different events related to the users' reading habits; Novel non-intrusive approach for active and continuous re-authentication; It took into account the security of the mobile application; The model is capable of accessing control, intrusion detection and recommender systems; The models test the authentication after x scrolls	The model should be the model changed to test for authentication after x amount of scrolls; Not evaluated on different test subjects; Dataset is scarcely described
Xu et al. [51]	Fourteen individuals carried in the research test. Every individual uses thirty minutes in every context. Moreover, the participants spend two to five minutes in each required activity under every context	An activity recognition method using contextual online learning techniques using data collected by low-cost inertial sensors is presented	Smartphones	No	The proposed algorithm outperforms the existing algorithms without requiring training phase from the individual; The algorithm is rigorously characterized; The method is capable of performing the recognition of activities with an online, personalized and adaptive method	The use of context did not significantly improve the results; Not evaluated with different test subjects and devices
Tang et al. [52]	22 individuals	The authors have tested ten classifiers on two public datasets for user activity recognition.	Mobile Devices	No	The authors used two published datasets with different activities; The datasets are clearly described; The authors tested different artificial intelligence methods to identify walking patterns	The authors should extend the dataset with other sensors; Not evaluated with different test subjects and devices
Yanyun et al. [53]	Thirty individuals. The test has used a total of 169 features. 75 horizontal and 22 vertical acceleration features; 22 triaxial acceleration magnitude features; 22 gyroscope and 22 geomagnetic features; and 6 atmospheric pressure features	A transportation mode recognition method based on Convolutional Neural Networks is presented	Smartphones	Yes	Convolutional Neural Networks used automatically learns the different transportation modes; It was performed the shallow feature extraction before the learning deep feature; The implemented method reduces the complexity; It is useful for mobile devices; A small number of layer works well for the pre-processing of the data	The authors should consider context-aware technologies; Not evaluated with different test subjects and devices
Li et al. [54]	Not mentioned	Proposes a lightweight indoor, outdoor and Semi-open recognition algorithm	Smartphones	No	It may realize localization indoor to outdoor, and vice versa; It presents the workflow of the algorithm implemented; The authors customized some methods to improve the energy efficiency	The authors should incorporate run-time verification methods to improve the accuracy and safety of the proposed method; Not evaluated with different test subjects and devices

Table 2. *Cont.*

Authors	Population	Purpose of Study	Device Type	Public Dataset	Pros	Cons
Yanjun et al. [55]	Eight individuals	Proposes a traffic pattern recognition method based on the Bayesian algorithm to identify the traffic	Smartphones	Yes	The algorithm used the sensors available in the mobile devices to detect traffic patterns; 400 h of data; The data was collected from different cities in the world; The transportation modes are presented	The proposed method presents very low accuracy when using only the acceleration sensor data. In this case, the distinction between the car and bus cannot performed; Not evaluated with different test subjects and devices
Vafeiadis et al. [56]	Three individuals	Proposes an occupancy detection algorithms using machine learning. The dataset consists of water and energy consumption information collected from smart meters	Mobile Devices	Yes	The use of smart meters for different sensing; The use of feature selection to discover the most important features; The authors used real-time data; The authors implemented several machine learning methods; It was used the boosting technique; The activities/environments are identified; Three individuals acquired data during one month; Ensemble classifiers achieved better performed than others, because it combines different classifiers; Feature selection helped to reduce the dataset sparsity	The authors must improve the dataset and the features used as input to the classifiers. Feature selection methods must be incorporated to enhance the accuracy of the predictive model.
Subasi et al. [57]	Seventeen participants have performed 33 fitness activities and use nine inertial sensors	Proposes an activity recognition method which uses data collected from sensors located on the body	Smartphones	Yes	The authors used the 10-fold cross-validation to test the algorithm; 33 fitness activities; The methods and workflow are clearly described	The study uses a high number of sensors; the authors must implement an attribute selection method due to the use of numerous sensors resulting in 117 attributes for each instance; Not evaluated with different test subjects and devices
Yuan et al. [58]	Not mentioned. The dataset consists of more than 36000 samples	Proposes an indoor localization method using mobile sensors	Mobile Devices	Yes	The authors implemented different combinations of Adaboost method; 36000 samples collected in a real-world environment; The features are clearly described; The proposed method outperforms the state-of-the-art, presenting low errors	The correlation between position x and y in the same location is not performed by the authors to improve accuracy; Not evaluated with different test subjects and devices

3. Results

As pictured in Figure 1, we identified 151 papers with three duplicates, that were removed. The other 148 articles were evaluated according to the title, keywords, and abstract, excluding 133 citations. After full-text evaluation, two papers were removed from the remaining 15 papers. The qualitative and quantitative synthesis included information related to the remaining 13 articles. In conclusion, we examined 13 documents.

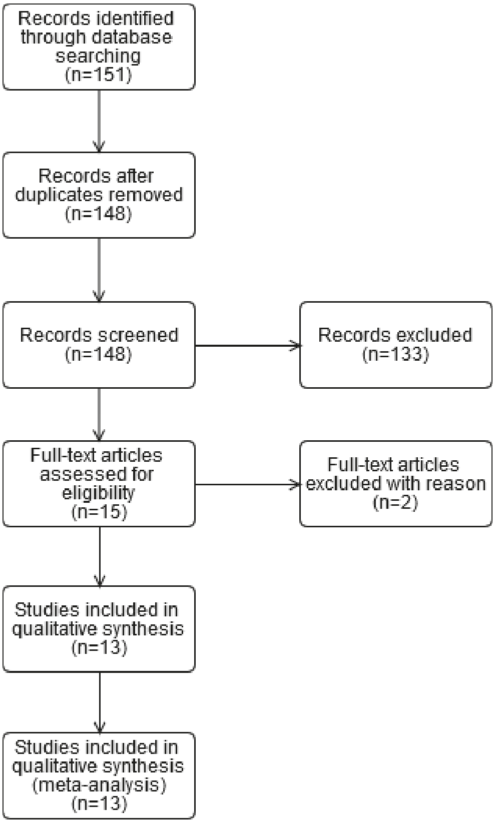

Records identified through database searching (n=151)
Records after duplicates removed (n=148)
Records screened (n=148)
Records excluded (n=133)
Full-text articles assessed for eligibility (n=15)
Full-text articles excluded with reason (n=2)
Studies included in qualitative synthesis (n=13)
Studies included in qualitative synthesis (meta-analysis) (n=13)

Figure 1. Articles analysis.

To find relevant information about the implementations presented in the different studies analysed in this review, the reader should find the information in the original cited works. Table 1 shows the year of publication and the resume of the papers and final results. Table 2 shows the population, the purpose of the study, devices, settings of the papers, pros, and cons. When the datasets used in a study is publicly available, or the population information is provided, it is considered as a positive aspect. In many cases, the evaluation uses a cross-validation scheme (regular or stratified per class). However, the studies do not consider different subsets of the population for training and testing (i.e., train/test split based on subjects or patients). This is generally a more rigorous evaluation scheme and is expected to hurt the reported accuracy. Other more specific pros and cons are provided for each study.

The papers were published between 2012 and 2018, where two studies were published in 2018 (15%), four studies were published in 2017 (31%), two studies were published in 2016 (15%), two studies were published in 2015 (15%), two studies were published in 2014 (15%), and one study was published in 2012 (8%). Regarding the used devices, it was split among 43% for smartphones and the remaining 57%

for mobile devices. The source code is not available for all studies analysed. Moreover, 69% of the studies have the raw data available. Finally, we verified that there are no studies that shared the source code.

Methods for Identification of Activities in Daily Living

In the study [57], the authors tried to use different classifiers for the recognition of activities with sensors to find the best method. Ten classifiers were utilized with the AdaBoost method. The dataset used was publicly available. The settings were investigated using nine inertial sensors from seventeen individuals taking into account 33 fitness activities. The used sampling rate was 50 Hz. After checking accuracies of the AdaBoost method, authors came to conclude that its implementation with random forest gives the best accuracy, with a value of 99.98%.

Authors of [49] have proposed harmonized authentication based on ThumbStroke dynamics (HATS) for mobile devices. The performance of HATS was tested, taking into account the different screen sizes of several mobile devices. Laboratory experiments were conducted to collect data for testing. Participants were required prior experience with touch screen devices and a qwerty keyboard. The study selected some features for learning ThumbStroke models, and these are timing features, spatial features, movement direction features, and operation features. The phrases, entered by the participants, were adopted from MacKenzie and Soukoreff and varied from 16 to 43 characters. Based method across all settings and classification models, the final results showed that HATS outperformed the keystroke dynamics. Among all the classification methods used, AdaBoost reported a maximum accuracy of 41.8%.

Li et al. [54] talks about an indoor/outdoor detection system (IOS). This method is split by the machine learning-based IOS-detector and the lightweight WiFi sub-detector. The first part infers indoor, outdoor, or semi-open environments based on the classification results. The second part focuses on the implementation of mobile devices. Finally, the other part consists of the IOS detection that shows high accuracy for the system. In conclusion, the proposed IOS detector achieves around 96% for the aggregated IOS detector and over 85% accuracy for the lightweight WiFi-based sub-detector.

In the study [50], the authors introduce a method for re-authenticating users taking into account a behavioral biometric-based on users' document scrolling traits. More specifically focused on identifying abnormal scrolling behavior on users while interacting with protected or read-only documents. Dataset was obtained from a previous project aimed to detect document access activities that indicate cyber attacks. Features for this paper were slit in vectors, being vector one derived from scrolling traits, vector two a representation of the polarity of scrolling, and vector 3 treats the dataset as a bipartite graph with two node sets. k-means clustering achieved the best performance with an 83.5% success rate in predicting the authenticated user.

The paper [48] presents a highly efficient method for the automatic detection of asthmatic wheezing in breathing sounds. The process is suitable for personal asthma monitoring via mobile devices since its not computationally complex. Most of the used data came from online databases of Human lung sounds. However, the authors also used several of their recordings of regular and wheezy breaths. The authors also confirmed the optimality of the audio spectral envelope (ASE) plus the value of the tonality index (TI) as a feature detector, using the mRMR (minimal redundancy–maximal relevance) method. Thousands of experiments were performed, and the best results were obtained from the fluctuation of the Audio Spectral Envelope descriptor adopted from the MPEG-7 standard, reporting an accuracy around 100%.

Authors of [53] developed a method to collect the sensor data, acceleration, gyroscope, geomagnetic, and atmospheric pressure were the four kinds of sensors used. The shallow feature extraction of the raw data happens before the CNN learning deep feature, which will reduce the complexity of the network and training time of the model. This process is critical for smartphones because of their limited resources. Three classes of features are extracted from each frame, including statistical, time, and frequency domains. Namely, the features used are: Mean, standard deviation, variance, median, minimum, maximum, range, interquartile range, kurtosis, skewness, root mean square, integral, double integral, autocorrelation, mean-crossing rate, fast Fourier transform, spectral energy, spectral entropy, spectrum peak position, wavelet entropy, and wavelet magnitude. Final results show that the proposed method can achieve 98% accuracy, meaning it outperforms the SVM (support vector machine) and AdaBoost classification in efficiency and computational cost, reporting accuracy of 93.6% with AdaBoost.

Yuan et al. [58] propose an indoor localization system using sensors for smartphones and smartwatches. Over 36,000 samples of data were collected in a 185.12 m^2 real indoor environment by a user using two different devices. Looking with the experimental results, the authors concluded that Twi-AdaBoost outperforms the state-of-the-art indoor localization algorithms. The localization error of position x and y achieved was 0.387 m and 0.398 m, respectively. The used datasets include the features: Place ID, Timestamp, Accelerometer_X, Accelerometer_Y, Accelerometer_Z, MagneticField_X, MagneticField_Y, MagneticField_Z, X_Axis Angle (Pitch), Y_Axis Angle (Roll), Z_Axis, Angle (Azimuth), Gyroscope_X, Gyroscope_Y, and Gyroscope_Z, reporting an accuracy around 99%.

In the paper [55], a novel technique based on the Bayesian voting algorithm that can be used with low-power sensors for transportation mode detection is presented. The authors used a set of data that consists of 400 h from eight individuals. Five sensors were used, being those: Acceleration, gyroscope, geomagnetic, barometer, and base station obtain by using AdaBoost classification to improve the results. Besides, the Bias algorithm was used to extract the features to reduce the adaptive boosting feature dimensions and determine the critical factors for identifying different transportation modes. The features used are: Mean, standard deviation, variance, median, minimum, maximum, range, interquartile, kurtosis, skewness, root mean square, time integral, double integral, auto-correlation, mean-crossing rate, fast Fourier transform, spectral energy, spectral entropy, spectrum peak position, wavelet entropy, wavelet magnitude, peak volume, intensity, length, variance of peak features, peak frequency, stationary duration, stationary frequency. Taking into account the final results, authors concluded that their algorithm could supply and replace some traffic pattern recognition algorithms and fix the problem that different mobile phones have various sensors, reporting accuracy between 64.54% and 96.83%.

In [51], the authors presented a contextual multi-armed bandits (MAB) approach that enables activity classification. This method makes context adaptation, continuous online learning, and active learning. Since the cost of extracting specific features is very high, the authors decided to use side information as the context. Since features can be used as contexts, this is not a limitation for the project. The proposed algorithm with active learning outperformed the benchmark algorithms by an average of 35%, reporting, and accuracy between 70% and 85%.

Xu et al. [47] focuses on three challenges, including the ability to accurately detect context using sensors and machine learning. The selection of activities for classification is performed by using context, reducing the complexity and improving the accuracy, speed, and energy usage, and the ability for experts in prescribing sets of physical activities under different environments. The features used for the project were: kNN (k-Nearest Neighbor) with time, kNN with wireless media access control (MAC) address and signal strength, and AdaBoost with audio peak frequency, peak energy, average power, and total energy. These were extracted from raw sensor data using a java program implementing the IContextFeatureExtractor interface. The data used was acquired by 14 participants that carried an Android mobile phone, and four 9-DOF devices were placed on dominant wrists, knee, ankle, and mid-waist. Each subject performed every required activity under every context for 2–5 min. The data were split into training (30%) and testing (70%) sets. Authors concluded that despite the methodology demonstrating effectiveness, efficiency, and potential, a more extensive study needs to be performed to improve privacy, security, and user-friendliness, reporting accuracy between 59% and 100%.

In [56], the problem of occupancy detection in a domestic environment was studied using machine learning techniques and their boosting versions on a dataset collected from electricity and water consumption smart meters. These features were selected using the Mutual Information technique. The dataset contains energy and water consumption (during summer) time data of 1-minute resolution for 16 consecutive days. The features included in the used dataset were: Central power, refrigerator, television, washing machine, dryer, cold water-kitchen, hot water-kitchen, dishwasher-water and washing machine-water, reporting accuracy higher than 70%.

Authors of [52] evaluated ten representative classifiers in the identification of two available datasets. The first dataset consists of accelerometer readings of walking patterns from 22 participants. The second one contains activity and postural transition data collected from the accelerometer and magnetometer data acquired from 30 participants. For the Walking dataset, the authors split the data into fixed-width sliding windows with a 50% overlap and extract nine features from every window and scale the features to $[-1, 1]$. The authors obtained the mean, standard deviation, and median absolute deviation from the different axis of the sensors. The authors of the study already pre-processed the sensor signals by noise filter and partitioned the data into fixed-width sliding windows with a 50% overlap as well and constructed a 561-feature vector for every window. From those features, authors extracted 24 features, including mean, standard deviation from the different axis of body acceleration, gravity acceleration, jerk signals of body acceleration, angular velocity, and jerk signals of angular velocity. In conclusion, the authors reported an accuracy between 95.6% and 97.8%.

The study [46] focuses on using mobile devices for the detection of cardiovascular autonomic neuropathy. The authors concentrated on the task of the detection and monitoring of cardiovascular autonomic neuropathy. After all the studies, they concluded that best outcomes were obtained by the novel combined ensemble of AdaBoost and Bagging based on the J48 decision tree, reporting the highest accuracy of 94.53%.

4. Discussion

This review confirms that AdaBoost, and in general boosting ensemble methods, are reliable for the identification of daily activities. Several studies are not well described, and the source code of the algorithms are not publically available. The verification and reproducibility of the obtained results is not easily possible, because of the following reasons: Only some authors shared the datasets; in many cases, the methods are not explained well explained, in particular, the preprocessing of the datasets; and the hyper-parameter tuning is poorly described, or the exact algorithm parameters are not described.

The number of studies using the AdaBoost method for the recognition of daily activities is minimal, and the daily activities mainly recognized are the simple activities, including walking, running, walking upstairs and downstairs, and other quotidian activities.

Following our literature review, most of the analysed studies (85%) report the best results using AdaBoost methods. Only two studies (15%) presented in [49,58] have said that the AdaBoost based methods do not show the best results when compared with the other approaches for daily activities and environments recognition. Nevertheless, the authors of these studies still recognised the reliable applicability of the AdaBoost method for activity and environment recognition activities.

In summary, all reviewed works first perform a feature extraction step, which somewhat varies depending on the used sensor types. In cases of multiple sensors, or multi-channel sensors, the feature extraction is performed independently for each time series (i.e., channel or sensor). Generally, various statistical metrics, as listed in Table 3, are computed on the raw signal in the time domain, and rarely features are deriving from the frequency domain. Then, after the features are extracted from each sensor as a separate time series, the extracted features are fed into the classifiers. Very often, a systematic approach to feature extraction improves the accuracy [23].

The authors used different features, and the average accuracies obtained with them can be comparable. Table 3 presents the average accuracy of the various features extracted, verifying that the features that allow the recognition of daily activities with an accuracy higher than 90% are the mean, standard deviation, pitch, roll, azimuth and median absolute deviation of signal of motion sensors, and the mean of the signal of magnetic sensors.

Table 3. Average of the accuracy reported in the studies analysed, grouped by features.

Feature	Average reported accuracy with AdaBoost
mean of signal of magnetic sensors	99.0%
pitch, roll, and the azimuth of the signal of motion sensors	99.0%
median absolute deviation of the signal of motion sensors	96.7%
mean of signal of motion sensors	96.0%
standard deviation of the signal of motion sensors	90.1%
median, variance, minimum and maximum values, interquartile range, range, skewness, kurtosis, integral, double-integral, Root Mean Square (RMS), Fast Fourier Transform (FFT), spectral entropy, spectral energy, wavelet entropy, spectrum peak position and wavelet magnitude of signal of motion sensors	87.1%
scrolling traits and polarity of scrolling	83.5%
peak volume, intensity, variance of peaks, stationary duration and stationary frequency of the signal of motion sensors	80.6%
peak frequency of the signal of motion sensors	80.3%
peak energy, average power and total energy of signal of motion sensors	80.0%

Moreover, Table 4 presents the advantages and disadvantages of the Adaboost method, proving that it can be used for the recognition of daily activities and environments with the recent advancements in the hardware and software of the devices commonly used.

Table 4. Advantages and disadvantages of the use of Adaboost method in the different studies analyzed.

Pros	Cons
- The combination of the Adaboost and J48 decision tree revealed the best results. - Adaboost can be used for the monitoring of diabetes. - Adaboost with Bagging and Boosting based on decision trees reported reliable accuracy. - This algorithm can be applied for real-time assessments with sensor data. - It provides high recognition accuracy and low computational complexity. - It provides high security and usability of the different implementations. - It can be executed in real-time with reliable accuracy. - The combination of Adaboost with the k-Nearest Neighbors algorithm outperformed all other classifiers. - The Adaboost method shows high reliability in the recognition of different activities. - The results obtained can be correlated between different devices.	- The research on multi-level classifiers should continue to improve the results. - The energy consumption of the Adaboost method is very high. - It should always have high reliability for medical purposes. - It has limited capabilities for recognition. - The classified should be updated with new data. - Larger-scale experiments need to be conducted to validate the efficacy of the algorithms further.

In comparison with other algorithms, the Adaboost method uses different algorithms as the weak learner, in which these algorithms will take into account the features extracted from the signals, such as mean, standard deviation, variance, and others. In general, Adaboost made use of complex data, but it can be used with 1D data in comparison with other algorithms. The authors of the research studies analysed used the Adaboost with uni-dimensional data, i.e., they used the features extracted from the data to provide the results, where the results obtained proved its reliability for physical and physiological data.

In conclusion, the use of mobile devices for daily activities recognition using AdaBoost is limited, because of the low power processing and battery capabilities of these devices [59,60]. According to the reported studies in this review, it is possible to conclude that the use of the AdaBoost method is reliable with mobile devices as verified by the accuracies reported in the different studies, where only two studies reported accuracies lower than 50%.

5. Conclusions

This review presents studies available in the literature that use the AdaBoost method for the recognition of daily activities and environments. Thirteen studies were analysed, and the main findings are summarised as follows:

- (RQ1) The AbaBoost method is an ensemble learning method that is used in conjunction with other algorithms. The different algorithms are commonly named as weak classifiers, avoiding the overfitting problem;
- (RQ2) The AdaBoost method is implemented in conjunction with other algorithms to increase the accuracy of the recognition of daily activities and environments;
- (RQ3) For the recognition of daily activities and environments, the AdaBoost method is combined with a weak classifier. The features that reported better accuracy are the mean, standard deviation, pitch, roll, azimuth, and median absolute deviation of the signal of motion sensors, and the mean of the signal of magnetic sensors.

This review also highlights the use of smartphones and other mobile devices as they should have a particular purpose because of limited battery life and processing capabilities. First, the authors excluded studies that are not focused on the recognition of daily activities end environments with the AdaBoost method. Secondly, the studies that do not use sensors available on mobile devices were excluded. We excluded several studies after analysis of the abstracts and full-text of the papers. Another reason for exclusion was the language of the study, excluding the studies that were not written in English. With the features collected, the AdaBoost method allows recognition with an accuracy higher than 80%.

As future work, the implementation of the AdaBoost method in the framework for the recognition of daily activities and environments; it will be used to recognize seven daily activities and nine environments.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation,writing—original draft preparation, writing—review and editing: J.M.F., I.M.P., G.M., N.M.G., E.Z., P.L., F.F.-R. and S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by FCT/MEC through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MEC através de fundos nacionais e quando aplicável cofinanciado pelo FEDER, no âmbito do Acordo de Parceria PT2020 no âmbito do projeto UIDB/EEA/50008/2020*).

Acknowledgments: This work is funded by FCT/MEC through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under the project **UIDB/EEA/50008/2020** (*Este trabalho é financiado pela FCT/MEC através de fundos nacionais e quando aplicável cofinanciado pelo FEDER, no âmbito do Acordo de Parceria PT2020 no âmbito do projeto UIDB/EEA/50008/2020*). This article is based upon work from COST Action IC1303-AAPELE-Architectures, Algorithms and Protocols for Enhanced Living Environments and COST Action CA16226-SHELD-ON-Indoor living space improvement: Smart Habitat for the Elderly, supported by COST (European Cooperation in Science and Technology). More information in www.cost.eu.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Schapire R.E. Explaining AdaBoost. In *Empirical Inference*; Schölkopf, B., Luo, Z., Vovk, V., Eds.; Springer: Berlin/Heidelberg, Germany, 2013.
2. Webb, G.I.; Zheng, Z. Multistrategy ensemble learning: Reducing error by combining ensemble learning techniques. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 980–991. [CrossRef]
3. Lorena, A.C.; De Carvalho, A.C.; Gama, J.M. A review on the combination of binary classifiers in multiclass problems. *Artif. Intell. Rev.* **2008**, *30*, 19. [CrossRef]

4. Ganjisaffar, Y.; Caruana, R.; Lopes, C.V. Bagging gradient-boosted trees for high precision, low variance ranking models. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 25–29 July 2011; pp. 85–94.
5. Lee, B.K.; Lessler, J.; Stuart, E.A. Improving propensity score weighting using machine learning. *Stat. Med.* **2010**, *29*, 337–346. [CrossRef] [PubMed]
6. Yang, P.; Hwa Yang, Y.; B Zhou, B.; Y Zomaya, A. A review of ensemble methods in bioinformatics. *Curr. Bioinform.* **2010**, *5*, 296–308. [CrossRef]
7. Zhou, Z.-H. *Ensemble Methods: Foundations and Algorithms*; CRC Press: Boca Raton, FL, USA, 2012.
8. Akhoundi, M.A.A.; Valavi, E. Multi-sensor fuzzy data fusion using sensors with different characteristics. *arXiv* **2010**, arXiv:1010.6096.
9. Banos, O.; Damas, M.; Pomares, H.; Rojas, I. On the use of sensor fusion to reduce the impact of rotational and additive noise in human activity recognition. *Sensors* **2012**, *12*, 8039–8054. [CrossRef] [PubMed]
10. Dernbach, S.; Das, B.; Krishnan, N.C.; Thomas, B.L.; Cook, D.J. Simple and complex activity recognition through smartphones. In Proceedings of the 2012 Eighth International Conference on Intelligent Environments, Guanajuato, Mexico, 26–29 June 2012; pp. 214–221.
11. Hsu, Y.; Chen, K.; Yang, J.; Jaw, F. Smartphone-based fall detection algorithm using feature extraction. In Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15–17 October 2016; pp. 1535–1540.
12. Paul, P.; George, T. An effective approach for human activity recognition on smartphone. In Proceedings of the IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, India, 20 March 2015.
13. Shen, C.; Chen, Y.; Yang, G. On motion-sensor behavior analysis for human-activity recognition via smartphones. In Proceedings of the 2016 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), Sendai, Japan, 29 Feburary–2 March 2016.
14. Wang, J.; Chen, Y.; Hao, S.; Peng, X.; Hu, L. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognit. Lett.* **2019**, *119*, 3–11. [CrossRef]
15. Gomaa, W.; Elbasiony, R. Comparative Study of Different Approaches for Modeling and Analysis of Activities of Daily Living. *SSRN Electron. J.* **2019**. [CrossRef]
16. Qi, J.; Yang, P.; Newcombe, L.; Peng, X.; Yang, Y.; Zhao, Z. An overview of data fusion techniques for Internet of Things enabled physical activity recognition and measure. *Inf. Fusion* **2019**, *55*, 269–280. [CrossRef]
17. Garcia, N.M.; Rodrigues, J.J.P. (Eds.) *Ambient Assisted Living*; CRC Press: Boca Raton, FL, USA, 2015.
18. Garcia, N.M. Roadmap to the Design of a Personal Digital Life Coach. In *International Conference on ICT Innovations*; Springer: Cham, Switzerland, 2015; pp. 21–27.
19. Sousa, P.S.; Sabugueiro, D.; Felizardo, V.; Couto, R.; Pires, I.; Garcia, N.M. mHealth sensors and applications for personal aid. In *Mobile Health*; Springer: Cham, Switzerland, 2015; pp. 265–281.
20. Dobre, C.; Mavromoustakis, C.X.; Garcia, N.M.; Mastorakis, G.; Goleva, R.I. Introduction to the AAL and ELE Systems. In *Ambient Assisted Living and Enhanced Living Environments*; Butterworth-Heinemann: Oxford, UK, 2017.
21. Felizardo, V.; Sousa, P.; Sabugueiro, D.; Alexandre, C.; Couto, R.; Garcia, N.; Pires, I. E-Health: Current status and future trends. In *Handbook of Research on Democratic Strategies and Citizen-Centered E-Government Services*; IGI Global: Hershey, PA, USA, 2015; pp. 302–326.
22. Goleva, R.I.; Garcia, N.M.; Mavromoustakis, C.X.; Dobre, C.; Mastorakis, G.; Stainov, R.; Chorbev, I.; Trajkovik, V. AAL and ELE Platform Architecture. In *Ambient Assisted Living and Enhanced Living Environments*; Butterworth-Heinemann: Oxford, UK, 2017; pp. 171–209.
23. Zdravevski, E.; Lameski, P.; Trajkovik, V.; Kulakov, A.; Chorbev, I.; Goleva, R.; Pombo, N.; Garcia, N. Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering. *IEEE Access* **2017**, *5*, 5262–5280. [CrossRef]
24. Aguileta, A.A.; Brena, R.F.; Mayora, O.; Molino-Minero-Re, E.; Trejo, L.A. Multi-Sensor Fusion for Activity Recognition—A Survey. *Sensors* **2019**, *19*, 3808. [CrossRef] [PubMed]
25. Esmaeili Kelishomi, A.; Garmabaki, A.H.S.; Bahaghighat, M.; Dong, J. Mobile User Indoor-Outdoor Detection through Physical Daily Activities. *Sensors* **2019**, *19*, 511. [CrossRef] [PubMed]
26. Deep, S.; Zheng, X.; Karmakar, C.; Yu, D.; Hamey, L.; Jin, J. A Survey on Anomalous Behavior Detection for Elderly Care using Dense-sensing Networks. *IEEE Commun. Surv. Tutor.* **2019**. [CrossRef]
27. Qolomany, B.; Al-Fuqaha, A.; Gupta, A.; Benhaddou, D.; Alwajidi, S.; Qadir, J.; Fong, A.C. Machine Learning, Big Data, And Smart Buildings: A Comprehensive Survey. *arXiv* **2019**, arXiv:1904.01460.

28. Nweke, H.F.; Teh, Y.W.; Mujtaba, G.; Al-Garadi, M.A. Data fusion and multiple classifier systems for human activity detection and health monitoring: Review and open research directions. *Inf. Fusion* **2019**, *46*, 147–170. [CrossRef]
29. Moustaka, V.; Vakali, A.; Anthopoulos, L.G. A systematic review for smart city data analytics. *ACM Comput. Surv. (CSUR)* **2019**, *51*, 103. [CrossRef]
30. Nweke, H.F.; Teh, Y.W.; Mujtaba, G.; Alo, U.R.; Al-garadi, M.A. Multi-sensor fusion based on multiple classifier systems for human activity identification. *Hum.-Cent. Comput. Inf. Sci.* **2019**, *9*, 34. [CrossRef]
31. Eakin, P. Problems with assessments of activities of daily living. *Br. J. Occup. Ther.* **1989**, *52*, 50–54. [CrossRef]
32. Law, M. Evaluating activities of daily living: Directions for the future. *Am. J. Occup. Ther.* **1993**, *47*, 233–237. [CrossRef]
33. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. From Data Acquisition to Data Fusion: A Comprehensive Review and a Roadmap for the Identification of Activities of Daily Living Using Mobile Devices. *Sensors* **2016**, *16*, 186. [CrossRef]
34. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Identification of activities of daily living using sensors available in off-the-shelf mobile devices: Research and hypothesis. In Proceedings of the International Symposium on Ambient Intelligence, Seville, Spain, 1–3 June 2016; pp. 121–130.
35. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S. Pattern recognition techniques for the identification of Activities of Daily Living using mobile device accelerometer. *arXiv* **2017**, arXiv:1711.00096.
36. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Goleva, R.; Zdravevski, E. Recognition of activities of daily living based on environmental analyses using audio fingerprinting techniques: A systematic review. *Sensors* **2018**, *18*, 160. [CrossRef] [PubMed]
37. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S. Approach for the development of a framework for the identification of activities of daily living using sensors in mobile devices. *Sensors* **2018**, *18*, 640. [CrossRef] [PubMed]
38. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C. Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices. *Pervasive Mob. Comput.* **2018**, *47*, 78–93. [CrossRef]
39. Pires, I.M.; Teixeira, M.C.; Pombo, N.; Garcia, N.M.; Flórez-Revuelta, F.; Spinsante, S.; Goleva, R.; Zdravevski, E. Android Library for Recognition of Activities of Daily Living: Implementation Considerations, Challenges, and Solutions. *Open Bioinform. J.* **2018**, *11*, 61–88. [CrossRef]
40. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Framework for the Recognition of Activities of Daily Living and their Environments in the Development of a Personal Digital Life Coach. In Proceedings of the DATA 2018: 7th International Conference on Data Science, Technology and Applications, Setubal, Portugal, 26–28 July 2017; pp. 163–170.
41. Pires, I.M.S. Multi-Sensor Data Fusion in Mobile Devices for the Identification of Activities of Daily Living. Ph.D. Thesis, Universidade da Beira Interior, Covilha, Portugal, 2018.
42. Pires, I.M.; Marques, G.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F.; Spinsante, S.; Teixeira, M.C.; Zdravevski, E. Recognition of Activities of Daily Living and Environments Using Acoustic Sensors Embedded on Mobile Devices. *Electronics* **2019**, *8*, 1499. [CrossRef]
43. Zdravevski, E.; Lameski, P.; Mingov, R.; Kulakov, A.; Gjorgjevikj, D. Robust histogram-based feature engineering of time series data. In Proceedings of the 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), Lodz, Poland, 13–16 September 2015; pp. 381–388.
44. Zdravevski, E.; Risteska-Stojkoska, B.; Standl, M.; Schulz, H. Automatic machine-learning based identification of jogging periods from accelerometer measurements of adolescents under field conditions. *PLoS ONE* **2017**, *12*, e0184216. [CrossRef]
45. Lameski, P.; Zdravevski, E.; Mingov, R.; Kulakov, A. SVM Parameter Tuning with Grid Search and Its Impact on Reduction of Model Over-fitting. In *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*; Springer: Cham, Switzerland, 2015; pp. 464–474.
46. Kelarev, A.V.; Stranieri, A.; Yearwood, J.L.; Jelinek, H.F. Empirical study of decision trees and ensemble classifiers for monitoring of diabetes patients in pervasive healthcare. In Proceedings of the 2012 15th International Conference on Network-Based Information Systems, Melbourne, Australia, 26–28 September 2012; pp. 441–446.

47. Xu, J.Y.; Chang, H.I.; Chien, C.; Kaiser, W.J.; Pottie, G.J. Context-driven, prescription-based personal activity classification: Methodology, architecture, and end-to-end implementation. *IEEE J. Biomed. Health Inform.* **2013**, *18*, 1015–1025. [CrossRef]
48. Wiśniewski, M.; Zieliński, T.P. Joint application of audio spectral envelope and tonality index in an e-asthma monitoring system. *IEEE J. Biomed. Health Inform.* **2014**, *19*, 1009–1018. [CrossRef]
49. Zhou, L.; Kang, Y.; Zhang, D.; Lai, J. Harmonized authentication based on ThumbStroke dynamics on touch screen mobile phones. *Decis. Support Syst.* **2016**, *92*, 14–24. [CrossRef]
50. El Masri, A.; Wechsler, H.; Likarish, P.; Grayson, C.; Pu, C.; Al-Arayed, D.; Kang, B.B. Active authentication using scrolling behaviors. In Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS), Amman, Jordan, 7–9 April 2015; pp. 257–262.
51. Xu, J.; Song, L.; Xu, J.Y.; Pottie, G.J.; Van Der Schaar, M. Personalized active learning for activity classification using wireless wearable sensors. *IEEE J. Sel. Top. Signal Process.* **2016**, *10*, 865–876. [CrossRef]
52. Tang, C.; Phoha, V.V. An empirical evaluation of activities and classifiers for user identification on smartphones. In Proceedings of the 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), Niagara Falls, NY, USA, 6–9 September 2016.
53. Yanyun, G.; Fang, Z.; Shaomeng, C.; Haiyong, L. A convolutional neural networks based transportation mode identification algorithm. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017.
54. Li, S.; Qin, Z.; Song, H.; Si, C.; Sun, B.; Yang, X.; Zhang, R. A lightweight and aggregated system for indoor/outdoor detection using smart devices. *Future Gener. Comput. Syst.* **2017**. [CrossRef]
55. Qin, Y.; Jiang, M.; Yuan, W.; Chen, S.; Luo, H. Transportation mode recognition algorithm based on Bayesian voting. In Proceedings of the 2017 5th International Conference on Enterprise Systems (ES), Beijing, China, 22–24 September 2017; pp. 260–269.
56. Vafeiadis, T.; Zikos, S.; Stavropoulos, G.; Ioannidis, D.; Krinidis, S.; Tzovaras, D.; Moustakas, K. Machine learning based occupancy detection via the use of smart meters. In Proceedings of the 2017 International Symposium on Computer Science and Intelligent Controls (ISCSIC), Budapest, Hungary, 20–22 October 2017; pp. 6–12.
57. Subasi, A.; Dammas, D.H.; Alghamdi, R.D.; Makawi, R.A.; Albiety, E.A.; Brahimi, T.; Sarirete, A. Sensor Based Human Activity Recognition Using AdaBoost Ensemble Classifier. *Procedia Comput. Sci.* **2018**, *140*, 104–111. [CrossRef]
58. Yuan, Y.; Melching, C.; Yuan, Y.; Hogrefe, D. Multi-device fusion for enhanced contextual awareness of localization in indoor environments. *IEEE Access* **2018**, *6*, 7422–7431. [CrossRef]
59. Pires, I.M.; Garcia, N.M.; Pombo, N.; Flórez-Revuelta, F. Limitations of the Use of Mobile Devices and Smart Environments for the Monitoring of Ageing People. In Proceedings of the ICT4AWE, Funchal, Portugal, 22–23 March 2018; pp. 269–275.
60. Pires, I.; Felizardo, V.; Pombo, N.; Garcia, N.M. Limitations of energy expenditure calculation based on a mobile phone accelerometer. In Proceedings of the 2017 International Conference on High Performance Computing & Simulation (HPCS), Genoa, Italy, 17–21 July 2017; pp. 124–127.

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

Electronics Editorial Office
E-mail: electronics@mdpi.com
www.mdpi.com/journal/electronics

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland

Tel: +41 61 683 77 34
Fax: +41 61 302 89 18

www.mdpi.com

ISBN 978-3-03936-339-1