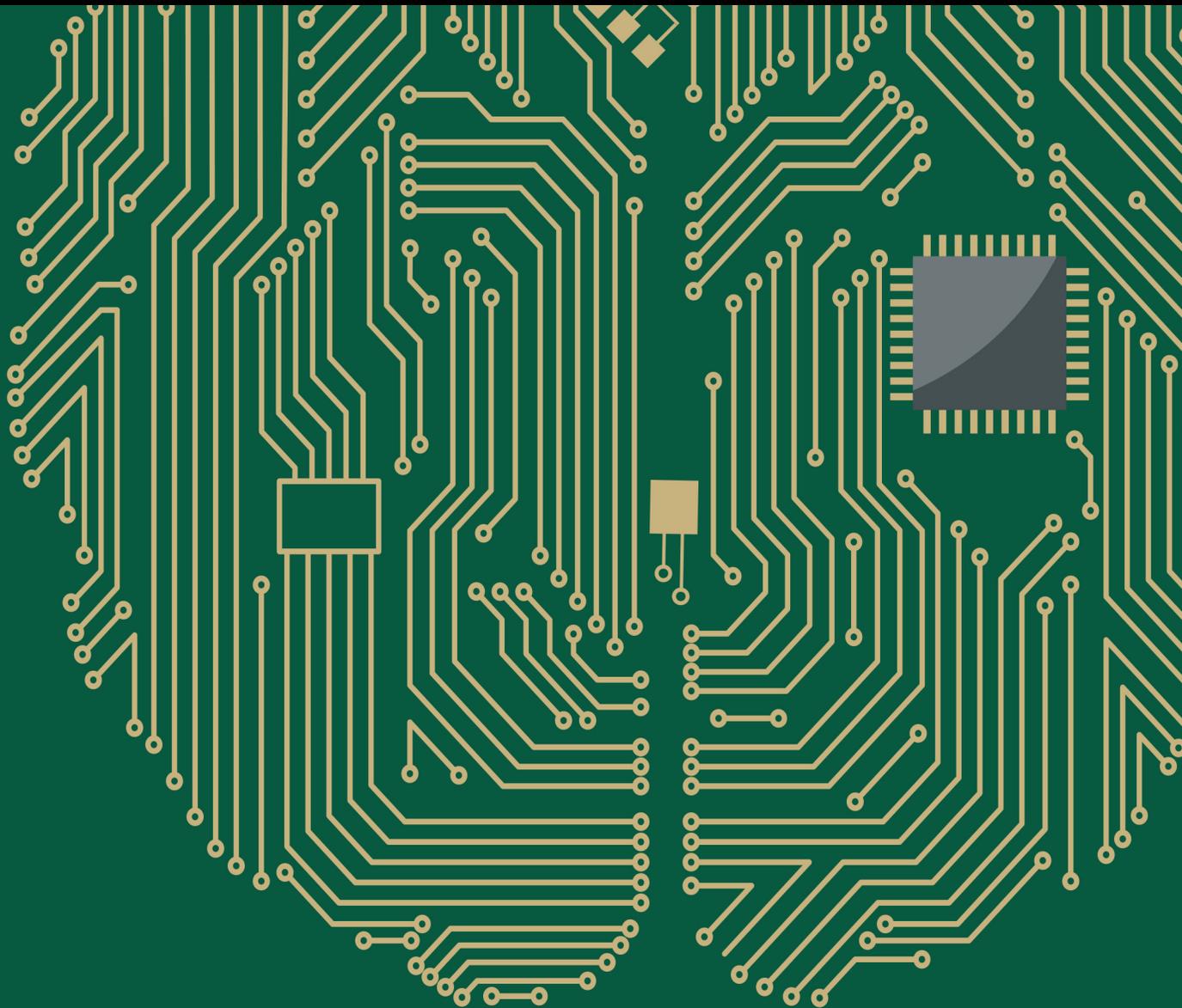


Brain-Inspired Intelligent Systems for Daily Assistance

Lead Guest Editor: Anastassia Angelopoulou

Guest Editors: Jose Garcia-Rodriguez, Epameinondas Kapetanios, Peter Roth,
and Kenneth Revett





Brain-Inspired Intelligent Systems for Daily Assistance

Computational Intelligence and Neuroscience

Brain-Inspired Intelligent Systems for Daily Assistance

Lead Guest Editor: Anastassia Angelopoulou

Guest Editors: Jose Garcia-Rodriguez, Epameinondas Kapetanios,
Peter Roth, and Kenneth Revett



Copyright © 2019 Hindawi. All rights reserved.

This is a special issue published in “Computational Intelligence and Neuroscience.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Editorial Board

Ricardo Aler, Spain
Amparo Alonso-Betanzos, Spain
Pietro Aricò, Italy
Hasan Ayaz, USA
Sylvain Baillet, Canada
Roman Bartak, Czech Republic
Theodore W. Berger, USA
Daniele Bibbo, Italy
Vince D. Calhoun, USA
Francesco Camastra, Italy
Michela Chiappalone, Italy
Andrzej Cichocki, Japan
Jens Christian Claussen, UK
Silvia Conforto, Italy
António D. P. Correia, Portugal
Justin Dauwels, Singapore
Christian W. Dawson, UK
Carmen De Maio, Italy
Sergio Decherchi, Italy
Paolo Del Giudice, Italy
Maria Jose del Jesus, Spain
Arnaud Delorme, France
Thomas DeMarse, USA
Anastasios D. Doulamis, Greece
Steven L. Fernandes, USA
Juan Carlos Fernández, Spain
Piotr Franaszczuk, USA
Leonardo Franco, Spain
Paolo Gastaldo, Italy
S. Ghosh-Dastidar, USA
Manuel Graña, Spain
Rodolfo E. Haber, Spain
Dominic Heger, Germany
Stephen Helms Tillery, USA
J. A. Hernández-Pérez, Mexico
Luis Javier Herrera, Spain
Etienne Hugues, USA
Ryotaro Kamimura, Japan
Pasi A. Karjalainen, Finland
Elpida Keravnou, Cyprus
Raşit Köker, Turkey
Dean J. Krusienski, USA
Fabio La Foresta, Italy
Antonino Laudani, Italy
Maciej Lawrynczuk, Poland
Mikhail A. Lebedev, USA
Cheng-Jian Lin, Taiwan
Giosuè Lo Bosco, Italy
Ezequiel López-Rubio, Spain
Bruce J. MacLennan, USA
Reinoud Maex, Belgium
Kezhi Mao, Singapore
Sergio Martinoia, Italy
Laura Marzetti, Italy
Elio Masciari, Italy
Paolo Massobrio, Italy
Gerard McKee, Nigeria
Michele Migliore, Italy
Paulo Moura Oliveira, Portugal
Debajyoti Mukhopadhyay, India
Klaus Obermayer, Germany
Karim G. Oweiss, USA
Massimo Panella, Italy
Fivos Panetsos, Spain
David M Powers, Australia
Sandhya Samarasinghe, New Zealand
Saeid Sanei, UK
Friedhelm Schwenker, Germany
Victor R. L. Shen, Taiwan
Fabio Solari, Italy
Jussi Tohka, Finland
Carlos M. Travieso-González, Spain
Lefteri Tsoukalas, USA
Pablo Varona, Spain
Roberto A. Vazquez, Mexico
Meel Velliste, USA
Mario Versaci, Italy
Francois B. Vialatte, France
Thomas Villmann, Germany
Ivan Volosyak, Germany
Cornelio Yáñez-Márquez, Mexico
Michal Zochowski, USA
Rodolfo Zunino, Italy

Contents

Anastassia Angelopoulou , Jose Garcia-Rodriguez , Epameinondas Kapetanios, Peter M. Roth, and Kenneth Revett

Editorial (2 pages), Article ID 7597839, Volume 2019 (2019)

Enhancing the Ambient Assisted Living Capabilities with a Mobile Robot

Francisco Gomez-Donoso , Félix Escalona , Francisco Miguel Rivas, Jose Maria Cañas , and Miguel Cazorla 

Research Article (15 pages), Article ID 9412384, Volume 2019 (2019)

Distinguishing Different Emotions Evoked by Music via Electroencephalographic Signals

Yimin Hou  and Shuaiqi Chen 

Research Article (18 pages), Article ID 3191903, Volume 2019 (2019)

Motor Imagery EEG Classification Based on Decision Tree Framework and Riemannian Geometry

Shan Guan , Kai Zhao , and Shuning Yang 

Research Article (13 pages), Article ID 5627156, Volume 2019 (2019)

Geoffrey: An Automated Schedule System on a Social Robot for the Intellectually Challenged

Edmanuel Cruz, Félix Escalona , Zuria Bauer, Miguel Cazorla , José García-Rodríguez , Ester Martinez-Martin , José Carlos Rangel, and Francisco Gomez-Donoso 

Research Article (17 pages), Article ID 4350272, Volume 2018 (2019)

A Novel Artificial Organic Control System for Mobile Robot Navigation in Assisted Living Using Vision- and Neural-Based Strategies

Hiram Ponce , Ernesto Moya-Albor , and Jorge Brieva 

Research Article (16 pages), Article ID 4189150, Volume 2018 (2019)

Editorial

Brain-Inspired Intelligent Systems for Daily Assistance

Anastassia Angelopoulou¹, **Jose Garcia-Rodriguez**², **Epameinondas Kapetanios**¹,
Peter M. Roth³ and **Kenneth Revett**⁴

¹University of Westminster, London, UK

²University of Alicante, Alicante, Spain

³Graz University of Technology, Graz, Austria

⁴HCL Infosystems Ltd., Boston, MA, USA

Correspondence should be addressed to Anastassia Angelopoulou; agelopa@westminster.ac.uk

Received 4 June 2019; Accepted 4 June 2019; Published 18 June 2019

Copyright © 2019 Anastassia Angelopoulou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The fields of machine learning and cognitive computing have been in the last decade revolutionised with neural-inspired algorithms (e.g., deep ANNs and deep RL) and brain-intelligent systems that assist in many real-world learning tasks from robot monitoring and interaction at home to complex decision-making about emotions and behaviours in humans and animals. While there are remarkable advances in these brain-inspired algorithms and systems, they need to be trained with huge data sets, and their results lack flexibility to adapt to diverse learning tasks and sustainable performance over long periods of time. To address these challenges, it is essential to gain an analytical understanding of the principles that allow biological inspired intelligent systems to leverage knowledge and how they can be translated to hardware for daily assistance and practical applications. This special issue brings researchers from interdisciplinary domains to report their latest research work on algorithms and neural-inspired systems that flexibly adapt to new learning tasks, learn from the environment using multimodal signals (e.g., neural, physiological, and kinematic), and produce autonomous adaptive agencies, which utilise cognitive and affective data, within a social neuroscientific framework. In this special issue, we have selected five papers out of fourteen high-quality papers after a careful reviewing process, which brings the acceptance rate to 35.7 percent. The five papers are representative of the current state-of-the-art in this area.

E. Cruz et al. present a robotic system for monitoring and interacting with people affected by cognitive diseases by successfully integrating object recognition, activity recognition,

localisation, and navigation methods to remember and help the patients to perform their daily tasks. The proposed methodology involves the implementation of an Object Recognition Engine (ORE) responsible to detect a certain object based on the Inception ResNet V2 architecture, a classic convolutional neural network (CNN) scheme over region-based convolutional neural network (R-CNN) architectures to avoid false detections. Furthermore, a Behaviour Recognition Engine (BRE) based on the OpenPose algorithm is then used to properly recognise the user's behaviours in different rooms and in different houses. The proposed system was tested successfully on the human-like robot Pepper where additional information was localised on the robot such as how the robot adapts to new environments (e.g., *furniture in a room has been rearranged*) based on a semantic localisation system (SLS) and how the robot navigates from one room to another based on expert systems that compute the path from the current room to the target. The proposed methodology shows competitive and consistent results.

On the contrary, H. Ponce et al. present a methodology inspired by nature-control systems for Robots in Assisted Living (RAL) navigation systems using vision-based strategies based on Hermite optical flow (OF) and convolutional neural networks (CNNs). This integrated system uses Hermite OF for obstacle motion detection and CNN for obstacle distance estimation. The authors estimate the distance to mobile and fixed objects using a monocular camera instead of RGB-D sensors that provide depth maps of the scene. The advantage of this method is that it needs less data for training the distance estimator and no training data to compute the OF

field. For experimental purposes, the authors used the robot simulator V-REP that recreates the conditions of the physical world quite accurately. The proposed methodology strengthens the hypothesis that a bioinspired OF method, a CNN technique for distance inference, and an artificial organic controller (AOC) can simulate a cognitive vision strategy in a dynamic environment.

S. Guan et al. propose first a subject-specific decision tree (SSDT) classification framework and second a data reduction method to distinguish multiclass motor imagery (MI) using electroencephalogram (EEG) signals for the brain-computer interface (BCI) based on the manifold of covariance matrices in Riemannian perspective. The goal of the SSDT classification framework is to separate the two MI tasks with the highest recognition rate and at the same time to enhance the classification accuracy. This is achieved by calculating in the tree a filter geodesic minimum distance to Riemannian mean (FGMDRM) in order to reduce the classification error. This method compared to other well-established methods in the literature performs better when EEG signals of the fixed frequency segment (8–30 Hz) are processed. The goal of the data reduction methodology, which includes a feature extraction and a classification algorithm, is to reduce in a nonlinear fusion the dimension of vectors in the Riemannian tangent plane and to classify different types of MI tasks based on the k-nearest neighbor (KNN) algorithm. The feature extraction algorithm, named “SJGDA,” combines semisupervised joint mutual information (semi-JMI) with general discriminant analysis (GDA). The proposed data reduction method performs better than semi-JMI and GDA on different datasets with higher recognition rates.

F. Gomez-Donoso et al. present a methodology that complements Ambient Assisted Living (AAL) environments, which are composed of cameras fixed to the ceilings of the environment, by integrating a domestic robot, namely, Pepper, which roams through the home to detect dangerous areas and movable (nonstatic) objects that the fixed cameras are not able to detect. In addition, the proposed system detects inhabitants’ fall event and their extended stay in a particular area of the home. The proposed pipeline implements three object detection and tracking algorithms. First, a multimodal evolutionary algorithm based on a set of single agents is used to detect and track people behaviour in 3D. Second, an Obstacle over the Ground Tracker (OGT) based on the Random Sample Consensus (RANSAC) model-fitting algorithm is coupled to the robot to assist it to detect fixed and moving objects and obstacles above the ground level. Finally, a Superficial Object Detector (SOD) algorithm is used to build a more comprehensive map of the potentially dangerous areas by recognising objects like wall or floor sockets that are too small to be sensed by the 3D camera. This is achieved by parsing the colour images captured by the camera of the robot to a region-based convolutional neural network (R-CNN) which returns the bounding box and the category of the objects. The performance of each algorithm is validated by a series of experiments in real-world environments.

Finally, Y. Hou and S. Chen present a system for characterising emotions using EEG signals, where in particular four classes of emotions (i.e., happy, sad, calm, and angry) should be distinguished. To this end, they induced these emotions by music stimuli (using 20 music passages in each music emotion group) and recorded the EEG signals of the subjects using 12 electrodes. From the obtained signals, 27 different features have been extracted, which have been used for further data analysis. In this way, the most valuable features have been identified by using the CFS method. In addition, it was revealed that C4.5 is more effective for emotion classification based on EEG signals than other widely used methods such as LDA or SVM.

Conflicts of Interest

The editors declare that there are no conflicts of interest of any kind regarding the publication of this special issue.

Anastassia Angelopoulou
Jose Garcia-Rodriguez
Epameinondas Kapetanios
Peter M. Roth
Kenneth Revett

Research Article

Enhancing the Ambient Assisted Living Capabilities with a Mobile Robot

Francisco Gomez-Donoso ¹, Félix Escalona ¹, Francisco Miguel Rivas,²
Jose Maria Cañas ² and Miguel Cazorla ¹

¹*Institute for Computer Research, University of Alicante, P.O. Box 99, 03080 Alicante, Spain*

²*RoboticsLab-URJC, Universidad Rey Juan Carlos, Madrid, Spain*

Correspondence should be addressed to Félix Escalona; felix.escalona@ua.es

Received 14 December 2018; Revised 26 February 2019; Accepted 10 March 2019; Published 2 April 2019

Academic Editor: Cornelio Yáñez-Márquez

Copyright © 2019 Francisco Gomez-Donoso et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ambient assisted living (AAL) environments are currently a key focus of interest as an option to assist and monitor disabled and elderly people. These systems can improve their quality of life and personal autonomy by detecting events such as entering potentially dangerous areas, potential fall events, or extended stays in the same place. Nonetheless, there are areas that remain outside the scope of AAL systems due to the placement of cameras. There also exist sources of danger in the scope of the camera that the AAL system cannot detect. These sources of danger are relatively small in size, occluded, or nonstatic. To solve this problem, we propose the inclusion of a robot which maps such uncovered areas looking for new potentially dangerous areas that go unnoticed by the AAL. The robot then sends this information to the AAL system in order to improve its performance. Experimentation in real-life scenarios successfully validates our approach.

1. Introduction

It is well known that ambient assisted living (AAL) environments will be a key feature of homes, offices, and even commercial facilities in the near future. AAL systems typically comprise several sensors and actuators installed in the home and some kind of communication that allows data to be gathered from them. Some intelligent processing of such data, in place or on cloud computing, may extract key information about the people living there, such as monitoring health or recognition of human activity. In addition, one of the most common tasks conducted by these systems is to automatically detect potentially dangerous and harmful areas of the environment such as nearby stairs, the kitchen, or the bathroom and send alerts to the persons living there and their relatives. AAL provides innovative approaches to the challenges of an aging population and physically or mentally challenged individuals, helping them to stay active longer, remain socially connected, and live independently into old age.

To do this, a number of cameras are usually placed on the ceiling of rooms in order to cover as broad an area possible using the minimum number of devices. An external system takes camera feed and is able to detect potentially harmful places, a person falling, or extended stays in a specific room, among other events. Nonetheless, smaller sources of danger such as electric panels and sockets around radiators or the oven remain unnoticed due to the placing of the cameras and the relatively small size of the mentioned elements. These zones are currently introduced manually into the system. Although it is not ideal, this is acceptable due to the fixed nature of the sources of danger, which will not change position over time.

However, there exist common threats like knives, a dog, or a robot vacuum cleaner which, in addition to their small size, are also nonstatic, i.e., their position could change over time. In these cases, it is not viable to manually set fixed zones of danger. Finally, it is worth noting that there will likely exist occluded areas caused by persons or furniture or

even the field of view of the camera. In this case, it is impossible to detect any sources of danger.

The main contributions of this work is the integration of a domestic social robot into the AAL system to detect the dangerous elements that cannot be detected by cameras alone due to their size, itinerant nature, or being located in an area that is occluded to the fixed cameras. The advantages and utility of the proposal are validated in actual office and home environments.

The rest of the paper is structured as follows: first, the state of the art in this field is reviewed in Section 2. Section 3 provides the proposed system description. Next, the experimentation is described in Section 4. Finally, the conclusions of this work are presented in Section 5.

2. Related Works

Ambient assisted living (AAL) is an emerging multidisciplinary field aiming to exploit information and communication technologies (ICT) in personal healthcare and telehealth systems to counter the effects of a growing elderly population [1]. Its aim is to help people to have an independent, monitored life with the use and assistance of technology.

AAL provides supportive home environments by integrating sensors, actuators, smart interfaces, and artificial intelligence [2]. There are interesting reviews of the AAL field [3–6], and several AAL frameworks and standardization efforts have been proposed such as ALLiance [7], Assisted Living Platform (ALIP) in the UK Dallas program, or Open Service Gateway Initiative (OSGi) platform. These aim to improve the interoperability and integration of medical devices in healthcare systems processing citizens' vital signs. However, currently there is no widely accepted de facto standard.

The sensors employed in AAL systems include wearable devices [8], Internet of things devices (IoT), and even cameras. Some of them share features of motion capture and human tracking systems [9, 10]. Specific health sensors like blood pressure or ECG are also occasionally considered. The use of IoT devices is an increasing trend [11–13]; they allow a ubiquitous, cheap, and flexible people monitoring. A good analysis can be found in [14].

Regarding communications, wireless systems like Bluetooth or Wi-Fi are prevalent. Low power technologies are usually preferred in this application domain.

In practical terms, fall detection is one of the most interesting situations to be monitored. In this context, there are several technological products in the market. The first are traditional monitoring systems such as pendants or wristbands worn by patients, who must activate these devices when needed, usually by pressing a button. The system sends an emergency call to the appropriate health service. These traditional systems require human intervention to report an alarm or ask for help, and user's potential noncompliance (both intended and unintended) can be a problem. In certain situations, for instance, if a patient falls to the floor as a consequence of fainting, he or she will not be able to activate the device, and this can be dangerous because the severity of

the damage may increase as time on the floor without health assistance increases. A second group of wearable systems relies on accelerometers and tilt sensors to automatically detect falls, but users may find continuously carrying these devices to be annoying.

Other solutions are embedded in the environment. They use external monitoring devices, and hence the user's compliance is not required. There are systems based on floor-vibrations, on infrared array detectors and on cameras. Within this broad area of possibilities, artificial vision provides a remarkably good sensor when developing applications for intelligent spaces. Cameras are passive sensors that supply a great amount of information and most are quite cost effective. However, a drawback to this solution is that they are sometimes seen as an invasion of the patient's privacy. Several vision based assistive systems use omnidirectional cameras.

Apart from AAL systems, the use of assistive robots as an aid in elderly daily lives is also an active research topic widely explored [15–17]. A good revision of robot assistants can be found in [18]. Some of them have been tested in laboratory scenarios, like Nao robot in [19], but in the most interesting proposals, the robots have been tested in real homes, even in long-term experiments. Hobbit robot [20] is a relevant case study. It is a care robot which is capable of fall prevention and detection as well as emergency detection and handling. Its interaction with the user is based on a multimodal user interface including automatic speech recognition, text-to-speech, gesture recognition, and a graphical touch-based user interface. Other interesting examples are the Max companion robot [21], several robotic platforms developed inside the EU FP7 Robot-Era Project [22], the GIRAFF robot [23], and the telepresence robot in [24].

Assistive robots may improve the quality of life of the elderly. They may help in bringing objects, in monitoring people's activities, as cognitive or emotional stimulation (like PARO robot [25]), providing reminders, providing telepresence to the medical professionals, detecting emergency situations, etc. Robots in healthcare are typically endowed with general robotic capabilities like autonomous navigation, manipulation, or perception but also with assistive specific capabilities. For instance, human robot interaction ability is very important in this application domain.

Many projects have also studied the use of robots in combination with AAL systems [26, 27]. For instance, Hendrich et al. [28] developed an AAL system that integrates service robots with sensor networks and user monitoring. The developed domestic robots there are based on the ROS middleware, and one of them has advanced manipulation capabilities. They also created the PEIS middleware that maintains the state of all sensors in the AAL environment, manages the high-level semantic information about objects and tasks, and provides the symbolic multirobot planner that controls the different robots, sensors, and smart appliances.

In addition, *user acceptance* is a hot topic in assistive robots as their usefulness heavily depends on the reactions they cause in the elderly final users. Many of the robots in healthcare case studies include a measurement and analysis of their user's acceptance. One broad comparative study of it

is provided in [29] with more than 70 robots in six month periods at user's home. It shows that users' evaluation of the robot dropped initially, but later rose after it had been used for a longer period of time. Moreover, some acceptance models [30] and acceptance measurement methodologies [23] have been proposed.

3. System Description

The proposed system aims to integrate a domestic robot into an AAL environment. As mentioned earlier, one of the main tasks of AAL is to detect potentially dangerous elements. However, there are a number of threats that will be unnoticed by the cameras of the system due to their relatively small size, or because they are in an occluded zone or are itinerant. In order to enhance the AAL system by making it aware of these threats, we propose the use of a mobile robot to detect potentially dangerous areas that the fixed cameras are not able to sense and thus improve the performance and robustness of the system.

There are two main elements in this approach. The first is the AAL environment which consists of a number of cameras fixed to the ceiling that are able to precisely localize the persons in the scene and to issue alerts when the person is in danger. The second is a robot that is in charge of continuously discovering new threats in the environment and sending them to the AAL system in order to incorporate these threats into its alert system. In the following subsections, both subsystems are described in detail.

The mobile robot of choice is Pepper. Pepper is a social robot manufactured by Softbank Robotics. It features a light-duty onboard computer which is able to perform simple tasks. It is able to move in planes, like the floor of a house, and also integrates color and depth cameras. We used the Pepper robot to implement our system, but any robot with color cameras, a mobile base, and a depth camera or laser could be used.

3.1. Ambient Assisted Living System. The ambient assisted living system proposed in this paper is based on people detection and tracking on 3D using depth sensors. Using the 3D information of the people tracked and their history of displacements in the room, the system can detect a set of risky situations. This system works 24/7 using depth information from an infrared sensor as the unique input. Figure 1 shows a scheme of the AAL system.

3.1.1. Background Subtraction. As input, the tracking algorithm expects only the foreground of the scene. That is why we need first to remove the background. This task was solved using a statistic background-learning algorithm based on a mixture of Gaussians but including a key improvement. The intensity of the noise of the sensor depends on the distance between the scene and the sensor itself. To reduce this effect, we introduce a logarithm-based image discretization and normalization (Figure 2). This reduces the distance given by the sensor in a set of bins where the longer

the distance, the greater each depth. Using this normalization, we can mitigate the effect of sensor noise.

3.1.2. Multimodal Evolutionary Algorithm for 3D People Tracking. A multimodal evolutionary algorithm has been designed to perform the detection and tracking behavior. This algorithm is based on a set of single agents:

- (i) Single: corresponds to the representation of a person in three dimensions, a prism with its position (x, y, z) and its size (dx, dy, dz) . This agent also contains information about the health and the subset of the associated point cloud.
- (ii) Races: a set of instances of the single agent. All of these try to explain a single agent. Multiple races can coexist inside the algorithm and each contains candidates for different races.
- (iii) Explorers: independent single agent. This agent tries to generate hints to allocate new races. A single explorer can generate a new race as long as there is enough evidence.
- (iv) Exploiters: members of a single race. Exploiters try to provide the most accurate explanation of a single agent.

The full pipeline of the algorithm is explained in Figure 3. The first step of the algorithm is to update existing races using evolutive techniques based on elitism and thermal mutation. Each race contains a series of exploiters from which the best candidate is chosen using a health function that combines density and completeness:

$$\text{health} = \text{density} + k * \text{completeness}. \quad (1)$$

The density of the health formula refers to the number of points per square meter inside its prim and the completeness is the ratio of points with respect to the exploiter of that race that has more points. The best exploiters are given thermal noise so that they evolve randomly over the areas that have been most successful and those with worst health are eliminated.

The next step is to create n "single" agents (explorers) on the zones that still contain data, once the information used to update the races has been removed. They are randomly generated all over the remaining point cloud. If any of these agents pass the criteria of similarity to the dimensions of a human and contains a minimum number of points, a new race will be generated. If there is a race with no associated data, it is also updated using a Kalman filter to estimate the evolution of the race.

3.2. Potentially Dangerous Areas. After the system is described, in this section, we explain what constitutes dangerous zone. The surroundings of a potentially harmful element are considered as dangerous zones. Elements such as electric panels or electric sockets that present a risk of a electrocution or shock, radiators or an oven that are burn hazards, and a robot vacuum cleaner or a dog that could make the person trip and fall, for instance, are considered

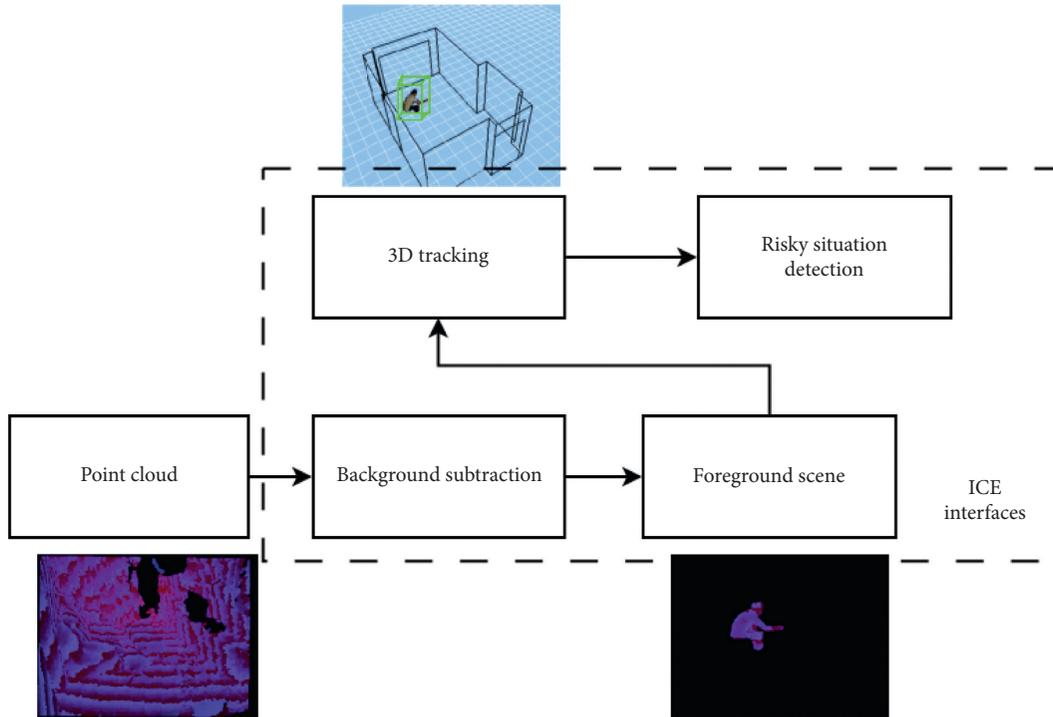


FIGURE 1: Block diagram of the AAL system including point cloud sensor, background subtraction, 3D object tracking, and risky situation detection.

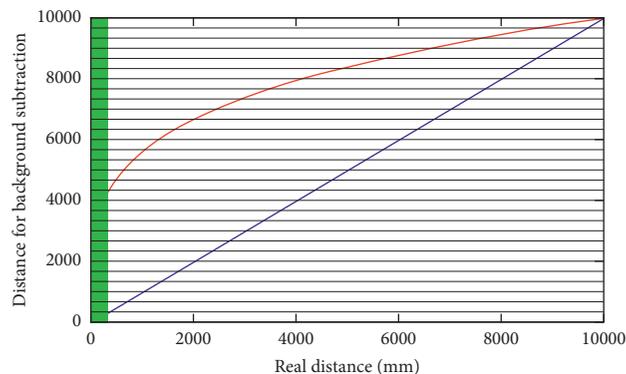


FIGURE 2: Depth discretization and normalization for background subtraction. The green area is where the sensor cannot capture information (object too close to the sensor).

dangerous and should be tracked by the system. In addition, the surroundings of items of furniture or, in essence, any element above ground level are also considered potentially dangerous areas.

This AAL is able to detect a set of risky situations based on the position of each person and their temporal trajectory:

- (i) Fall: the system is able to detect if a person has fallen in a supervised area. Once the situation has been detected, the system can wait a few seconds to verify whether there actually is a risky situation or simply trigger an automatic alarm.
- (ii) Dangerous areas: some areas inside the supervised area can be labeled as dangerous. If the AAL system detects a person close enough to any of these areas,

an alarm will be automatically triggered. This is useful when there is a balcony in the area or an exit door, for instance.

- (iii) Room entry or exit: AAL has a count of how many people are located in each area. This can be useful to detect entry into prohibited areas like the cleaning room or the medicine room of a nursing home.
- (iv) Absence: another risky situation can be the absence of a person from a certain room over a long period of time.
- (v) Extended stay: monitoring a toilet with any kind of sensor can generate controversy. In order to avoid including a sensor in this kind of room, the AAL system detects the entry of a person into a room

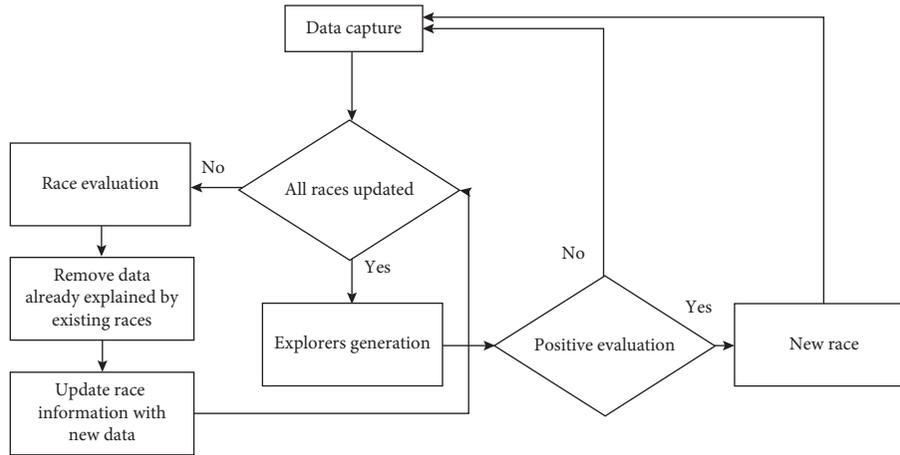


FIGURE 3: 3D multimodal evolutionary tracking algorithm.

without another exit and is able to trigger an alarm if the person does not return to the main room in a specific period of time.

As mentioned, some of these dangerous areas are easily detected by the AAL system, but there still exist harmful sources that cannot be detected. There are three kinds of potential threats that can remain unnoticed by the AAL cameras system:

- (i) Relatively small elements: due to the field of view of the cameras of the AAL system, some objects could be perceived as being relatively small in size. This will cause any trial processing of images to fail because there is insufficient visual information. Furthermore, they will likely not be sensed by the tridimensional camera at all. These types of objects include knives, razors, shoes, or a robot vacuum cleaner.
- (ii) Occluded elements: there could be areas of the environment that the cameras of the AAL system cannot cover. For instance, a dog might be behind a table, which is a tripping risk, and the cameras would not detect it.
- (iii) Itinerant elements: nonfixed elements also represent a set of dangerous objects, as they cannot be detected by the AAL system.

In order to make the system detect these areas and to issue a corresponding alert, we need an additional agent. We propose the use of a mobile robot, which would roam the environment searching for these threats.

3.3. Using a Robot for Dangerous Areas Detection. As explained earlier, there are a variety of situations in which the AAL may not detect dangerous objects and events. We propose the integration of a mobile robot into the AAL system that can detect these cases and send them to the AAL in order to improve its performance.

The mobile robot of choice must be equipped with color and depth cameras, and it is assumed that it can move in the

environment. The robot runs the pipeline depicted in Figure 4, which is described in detail in the following subsections.

3.3.1. Detection of Objects on the Ground. It is worth noting that all the objects above ground level could be a source of dangerous situations so must be considered by the AAL system. As explained earlier, the AAL system uses a static map of the environment which allows it to detect fixed obstacles like walls and doors. Nonetheless, moving obstacles are also a source of danger, so we propose the following algorithm to detect fixed and moving objects and obstacles above the ground level. This algorithm runs in the mobile robot.

First, the robot captures a color image and the corresponding depth map. Using both data streams a color point cloud is generated. Then, the resultant point cloud of the scene is transformed to the global robot coordinate frame T^* ; $T^* = [R^* | t^*]_{4 \times 4}$. At the start of the algorithm, if it is the first frame, the transformation is the identity $T^* = I$, namely, the initial robot coordinate frame is assumed as the global robot coordinate frame. If it is not the first frame, the current transformation T ; $T = [R | t]_{4 \times 4}$ is accumulated to the global transformation $T^* = T^* \times T$. To compute the current transformation T , we used robot odometry. In this way, each point cloud $Pc = \{P_i\}$; $P_i = (x, y, x, r, g, b)$ is transformed to the robot global coordinate frame $Pc^* = Pc \times T^*$, thus creating a tridimensional map of the environment $M = M + Pc^*$.

The next step is to detect the floor plane, so we used RANSAC [31] to carry out this process. RANSAC is a model-fitting algorithm that takes a set of data and tries to fit it in a model. In our case, the input data are the recently acquired point cloud Pc^* and the model is a plane. As a result, this step returns the coefficients a, b, c, d of all the detected planes in the scene. Each plane is modeled as $ax + by + cz = d$. A simple test, which consists of checking whether the z component is about 0 within a threshold, is carried out to reject the planes that are not at ground level. As the planes are estimated over data gathered by the sensor, it is likely that

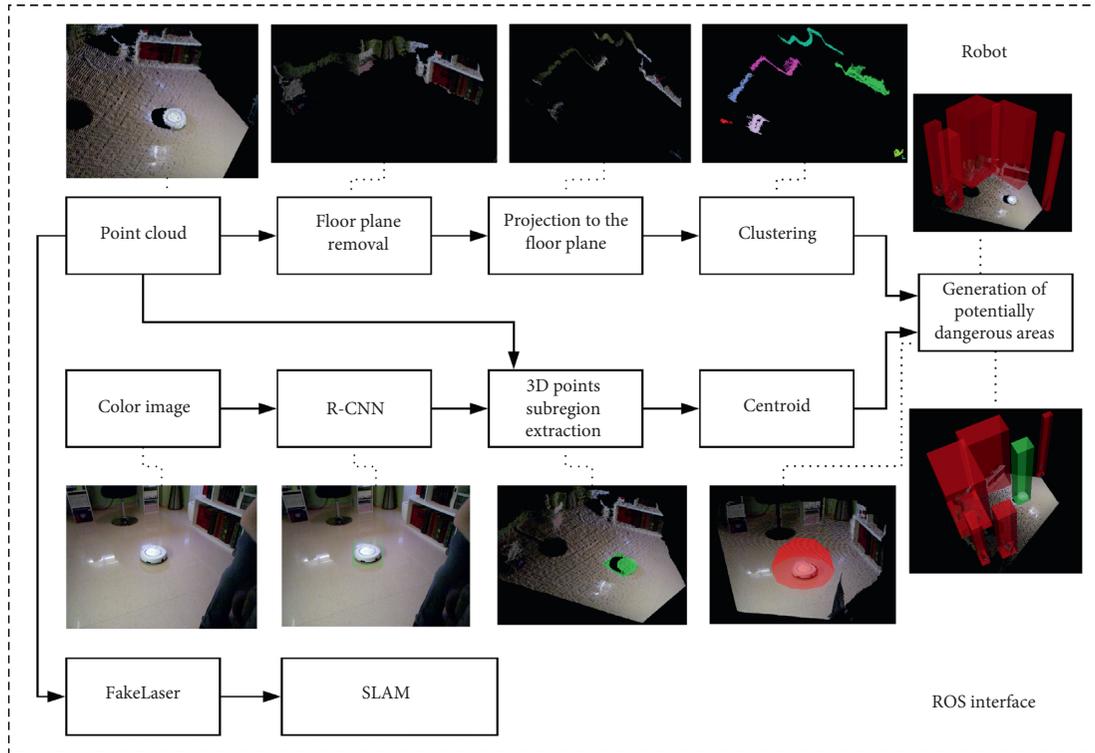


FIGURE 4: Description of the OGT and SOD algorithm pipelines which run in the robot.

the z component is not exactly 0, but very close to it. The points that lay in that approximate plane of the floor within a threshold are deleted from the point cloud. At this point, only the objects above ground level remain in the point cloud. This threshold prevents noise and possible artificial artifacts from being detected as obstacles. Then, the points of each object are projected to the previously computed ground plane. As a result, a 2D map of the obstacles above ground is obtained.

In the next step, a Euclidean clustering process is performed in order to segment and isolate the obstacles. The Euclidean clustering method essentially groups points that are close together. A cluster tolerance threshold is set so the points within this threshold are considered to be part of the same cluster. As a result, this process will return a cluster for each obstacle. A rectangle is then fitted to the points of each cluster. The rectangles are used to build the dangerous zones by extruding the potentially dangerous areas in the Z axis in order to transform the map to the tridimensional space once again. Note that the clusters are not intended to have a semantic meaning; we pursue the best possible geometrical fitting. Consequently, each cluster may not represent a specific object.

Finally, current dangerous areas are fused with previously detected dangerous areas if they overlap. This process is looped as the robot moves, thus building a map of the environment and keeping track of the potentially dangerous areas. We named this algorithm the Obstacles over the Ground Tracker (OGT).

3.3.2. Superficial Object Detector. Although the OGT algorithm performs reasonably well for large obstacles, some

dangerous areas will still not be detected, as stated before. For instance, objects like wall or floor sockets, electrical panels, or vacuum cleaners are not detected by the described algorithm because they are too small to be sensed by the 3D camera. Hence, we propose the following pipeline to detect these cases. This algorithm is executed alongside the OGT.

First, the color image captured by the camera of the robot is fed to a region-convolutional neural network (R-CNN). The R-CNN is able to return the bounding box and the category of the objects it detects. Then, the points inside each area of interest of the detected objects are extracted. This process is straightforward as the color image, and the point cloud is registered beforehand. Next, for each subset of points inside the bounding boxes, the median-centroid is computed. This is done due to the presence of nonobject points in the bounding boxes. As the bounding boxes are rectangles, the majority of the points belong to the object of interest, but there are still background points. A cube is then fitted for the points of each object but, this time, keeping the center of the cube in the previously computed median-centroid.

In this way, we can use the color information to detect these risks that are not sensed by the tridimensional sensor or ignored by the OGT and build a more comprehensive map of the potentially dangerous areas. We named this algorithm the Superficial Object Detector (SOD).

Finally, the potentially dangerous areas detected by the OGT and SOD pipelines running in the robot are merged and sent to the AAL system in order to detect whether the person enters one of these zones.

It is worth noting that there are likely to be potentially dangerous zones that could be detected by different systems

at the same time. For instance, walls are detected by the three methods. In addition, objects located on top of other objects are also redundantly detected. For instance, electric sockets in the walls or a knife on a table. The reason for not filtering these cases is twofold. First, the redundant detections improve the robustness of the system, and second, it could enhance the alerts emitted by the AAL system by adding semantic data.

3.4. 3D World References Registration between AAL and the Robot. As mentioned before, the AAL and the robot are continuously sharing information. For instance, the potentially dangerous objects detected by the robot are sent to the AAL. The robot transmits the tridimensional position of the objects in its own reference frame. Nonetheless, the event of a user trespassing the area near that object is detected by the AAL. Given this pipeline, this process can be only carried out if both systems are working in the same reference frame. This process is depicted in Figure 5.

It is worth noting that integrating two systems working on their own references and coordinates is not an easy task. In order to interact between systems, both need to be in the same system of coordinates with the same reference axis.

To solve this issue, both systems have been calibrated using the same coordinate reference. On one hand, the AAL detects a known pattern (both location and size are known, Figure 6(a)) and estimates its position using a Perspective-N-Point algorithm [32]. This way, the camera is located within the coordinate frame of the pattern. The robot follows the same procedure. It solves the Perspective-N-Point problem to locate itself in the coordinate frame stated by the pattern (Figure 6(b)). This calibration procedure enables both devices (AAL and robot) to share the same coordinate frame reference using the obtained transformation matrices T_A and T_{R_0} .

This calibration step is performed once at the setup stage. However, if the robot moves, the transformation T_{R_0} we previously computed is no longer valid. To solve this issue, we rely on SLAM methods.

The localization of the robot within the environment is carried out using SLAM algorithms. Specifically, it uses the GMapping ROS Package [33], which implements the Monte Carlo Localization algorithm. This algorithm uses laser scans as input. Nonetheless, the laser sensor of our robot is quite limited and noisy; we used the depth camera to simulate it. First, a depth map is captured using the aforementioned depth camera. Then, we extract the central row of the depth map. As the values of the depth map are in fact distances to the objects in the scene, the reinterpretation to laser scans is straightforward. This process is named “FakeLaser” in Figure 4. As a result, this method provides additional transformations T_{R_1}, \dots, T_{R_n} . This chain of transformations describes the position of the robot, so they are used to compute the transformation between the coordinate systems of the AAL and the robot even if it moves.

Summarizing, the AAL and the robot are both calibrated using a common pattern. As a result, both devices are localized in the same 3D coordinate frame through the transformation matrices T_A and T_{R_0} . This step allows the

transformation of the 3D objects detected by the robot to the AAL coordinate frame. If the robot moves, the transformation is no longer valid, so we rely on the mentioned SLAM algorithm to compute additional transformations T_{R_1}, \dots, T_{R_n} . The chain of transformations enables the transformation between the coordinate frame of the robot and the AAL even if the robot moves.

4. Experimentation

In this section, we describe the experimentation of the AAL, OGT, and SOD algorithms separately in order to validate the detection of potentially dangerous areas. Then, results of combining both pipelines and the AAL system are also presented.

It is worth noting that we used an AAL system provided by Pentalo Labs which features an Intel i3 powered NUC (i3-7100U) and an Asus Xtion sensor (Figure 7). This sensor provides RGB and depth information. We only use depth information for the core algorithm of the AAL system. RGB information is only used for visual validation. We also used a Pepper Robot as the mobile robot of choice. This robot is equipped with a color and a depth camera and is able to move and compute the transformation between two frames through self-odometry. Due to the limited computational power of the onboard processor, all the computation is executed on an auxiliary computer equipped with an Intel i5-3570 CPU, 16 GB DDR3 RAM, and an Nvidia 1080Ti GPU. The R-CNN implementation leverages the GPU for accelerated algorithms. Communication between devices is provided by ROS Kinetic [34], JdeRobot [35], and ICE [36]. The operating system of choice is Ubuntu 14.04.

To make the verification of the system as accurate as possible, a log system was developed. This system is able to record information from a set of devices and save all the information to the hard disk of the computer. Subsequently, the data can be replayed identically to how it was provided by the physical device. This procedure was applied in order to verify the precision of the situation detected in all the following experiments. Following this procedure, we can synthetically reproduce real daily life experiments including external perturbation to the data to verify the robustness of the algorithm. An automatic evaluator has been created to ensure the precision of the experiment. This evaluator will reproduce the recorded log (which contains certain risky situation) 3000 times with different kind of noise. If the expected risky situation has been detected, the test will be labeled as a success. Each test will require 3000 times the duration of the recorded log.

4.1. AAL Tested in Residential Environment. In this section, we put to test the AAL system in a residential environment.

The devices used in the residential environment consists of a single node with two depth devices (Figure 8(a)). The field of view of each device can be seen in Figures 8(b) and 8(c).

- (i) A fall situation seven meters from the device with a high occlusion level and covered only with one device (Figure 9(a))

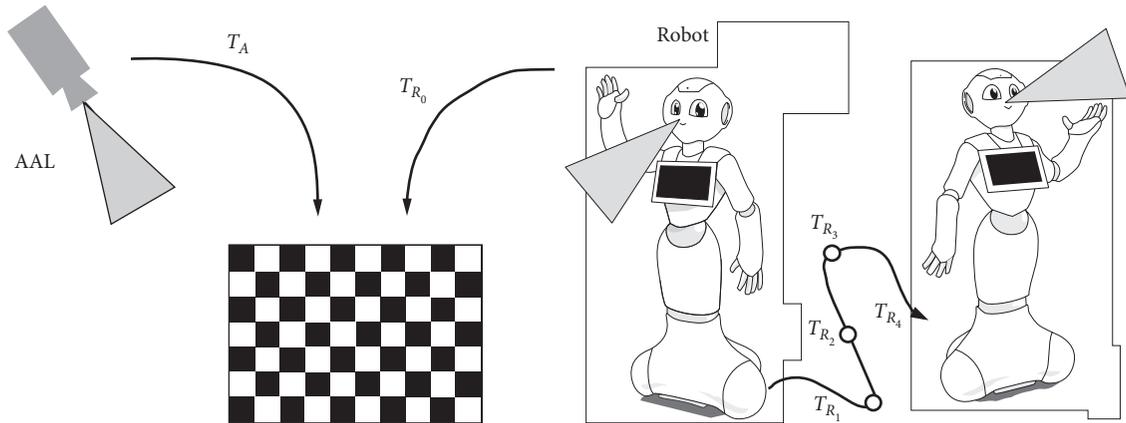


FIGURE 5: Diagram describing the method to compute the transformations between the AAL and the robot that enables the correct sharing of the tridimensional position of the detected potentially dangerous areas.



(a)



(b)

FIGURE 6: In order to integrate the potentially dangerous areas detected by the robot with the AAL system, both coordinate frames must be registered. We used a common pattern to calibrate both systems in the same coordinate frame.



FIGURE 7: Intel NUC with an Asus Xtion running the AAL system.

All the situations were correctly detected with a ratio of 100%. Each risky situation was evaluated 3000 times using a different level of noise disturbing the depth sensor of the system. The results of this experimentation are presented in Table 1.

4.2. AAL Tested in Clinical Environment. The second experiment focuses on a potential scenario where this system can also be applied, which is the clinical environment of a nursing home. In this case, a set of two devices and a single node were distributed following the scheme presented in Figure 10(a). The field of view of each sensor can be seen in Figures 10(b) and 10(c). Using this scheme, a set of four risky situations was recreated:

- (ii) A fall situation in the center of the room where the area is covered by both devices (Figures 9(b) and 9(c))
- (iii) A proximity risky situation near the door of the balcony (Figure 9(d))
- (iv) An extended stay situation where a detected person enters the toilet and does not return within ten minutes
- (i) A fall situation collected with a single device nine meters from the sensor (Figure 11(a))
- (ii) A second fall situation in the center of the room simultaneously recorded by both devices (Figures 11(b) and 11(c))
- (iii) A third fall situation very close to a window where the sun's infrared rays affected the device's performance

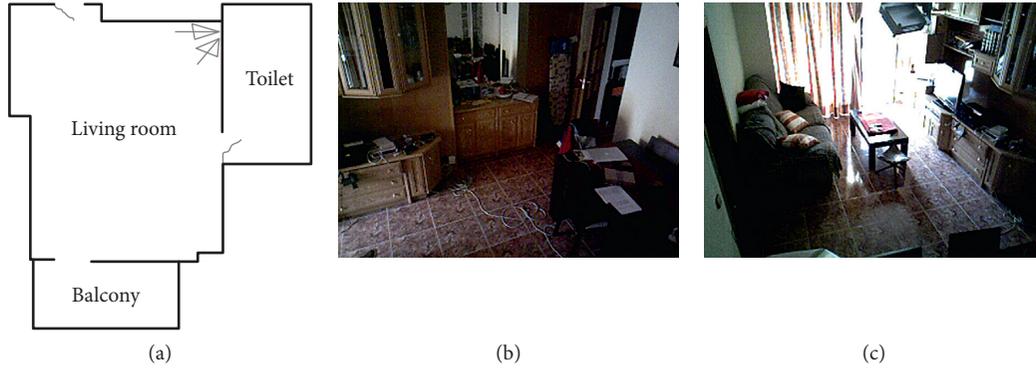


FIGURE 8: Location of devices and field of view.



FIGURE 9: Three different risky situations (a–c) (the same in two views simultaneously) and (d).

TABLE 1: Detection results across the residential environment.

Risky situation	True positive	False negative
Fall 1	3000	0
Fall 2	3000	0
Proximity situation	3000	0
Extended stay	3000	0

- (iv) An extended stay situation when a detected person enters the toilet but does not return within 10 minutes

All fall occurrences were properly detected in the 3000 repetitions of each situation regardless of the noise level, while the accuracy for the extended stay situation is 96%. With a high level of noise, these risky situations were not detected. During these failures, the system detected that the person left the room using the door. The door of the toilet is approximately 9 meters from the sensor, so the data are already very noisy without introducing extra noise. All situations without extra noise were correctly detected. The results of this experiment are shown in Table 2.

4.3. Robot Running the Objects on the Ground Tracker Algorithm Experiments. This algorithm takes the point cloud

provided by the Pepper Robot as input. The point clouds are dense with a 640×480 resolution, meaning there are 307200 points in each. The first step of the OGT algorithm is the removal of the floor plane, which is performed with RANSAC. The target model is a plane, and the inlier threshold is 3 mm. The projection to the floor plane is straightforward and has no parameters. The Euclidean clustering process rejects clusters with less than 200 points. This is done to filter smallish clusters that could emerge due to the noise present in the point cloud. In addition, cluster tolerance is 3 mm, which is the distance between two adjacent points that are in the same plane. All these parameters were set empirically.

The robot was deployed in an office environment. It was placed in the center of the room and performed a 360 degrees turn by rotating its base. The OGT algorithm was executed for each frame.

The results of this experiment are shown in Figure 12. As expected, the majority of the objects and obstacles were properly detected, and the potentially dangerous areas were created in the tridimensional map. Only three objects were not detected: a laptop power adapter, a knife, and a bottle. They were ignored by the OGT algorithm due to the floor plane tolerance threshold. The bottle was also placed on the

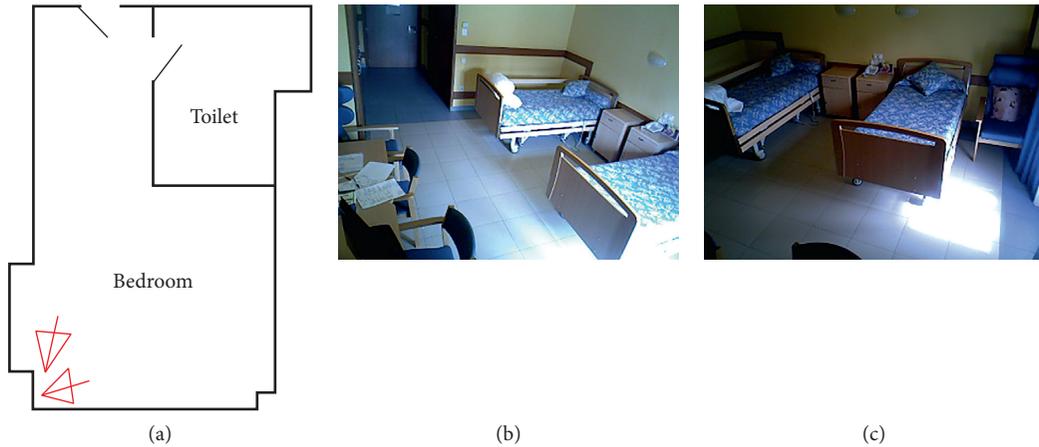


FIGURE 10: Location of devices and field of view.

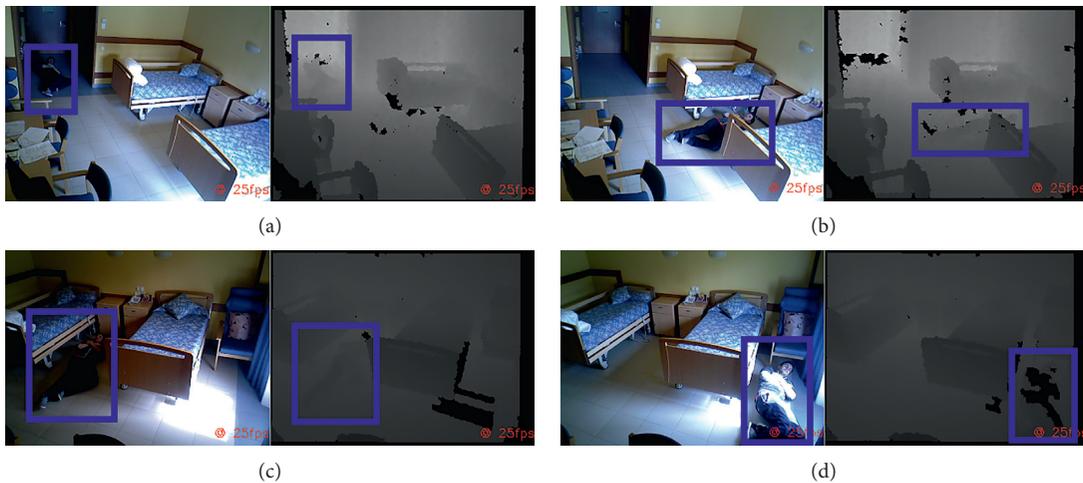


FIGURE 11: Three different risky situations (a-c) (the same in two views simultaneously) and (d).

TABLE 2: Detection results across the clinical environment.

Risky situation	True positive	False negative
Fall 1	3000	0
Fall 2	3000	0
Fall 3	3000	0
Extended stay	2881	119

floor, but its tridimensional information was removed due to its minimum cluster size. It is worth noting that due to the materials, the only part of the bottle sensed by the camera is the label. The plastic shape cannot be detected by the RGB-D sensor of the robot.

4.4. Robot Running the Superficial Object Detector Algorithm Experiments in Office Environment. As mentioned earlier, the color images and the point clouds are provided by the Pepper Robot at 640×480 resolution. The image and the point cloud are registered, so for each pixel in the color image, there is a corresponding point in the point cloud. Then, the color image is resized to 416×416 , which is the

input size of the YOLOv3 [37] architecture. YOLOv3-416 is the chosen R-CNN implementation. This version provides 35 fps on a Nvidia 1080Ti, which is suitable for real-time uses, while currently being one of the most accurate architectures. The output of this architecture is composed of the bounding boxes of the detected objects and their corresponding category.

Instead of taking an already trained model, we trained one from scratch. As mentioned before, the system must detect the dangerous items that are likely to appear in indoor environments. The categories of objects we selected are electrical panels, tangles of wires, wall and floor mounted sockets, knives, ovens, shoes, bottles, hobs, cats, dogs, and robot vacuum cleaners. In total, we consider 11 categories. In order to train the model, we built a custom dataset. To do so, 11,000 images were downloaded automatically from free stock images websites. The number of images per category is balanced, so the dataset comprehends 1,000 images per class. The images were automatically downloaded in bulk by searching for the keywords mentioned before. A team of 5 human agents curated the dataset by ensuring that the images correctly depicted the categories and manually

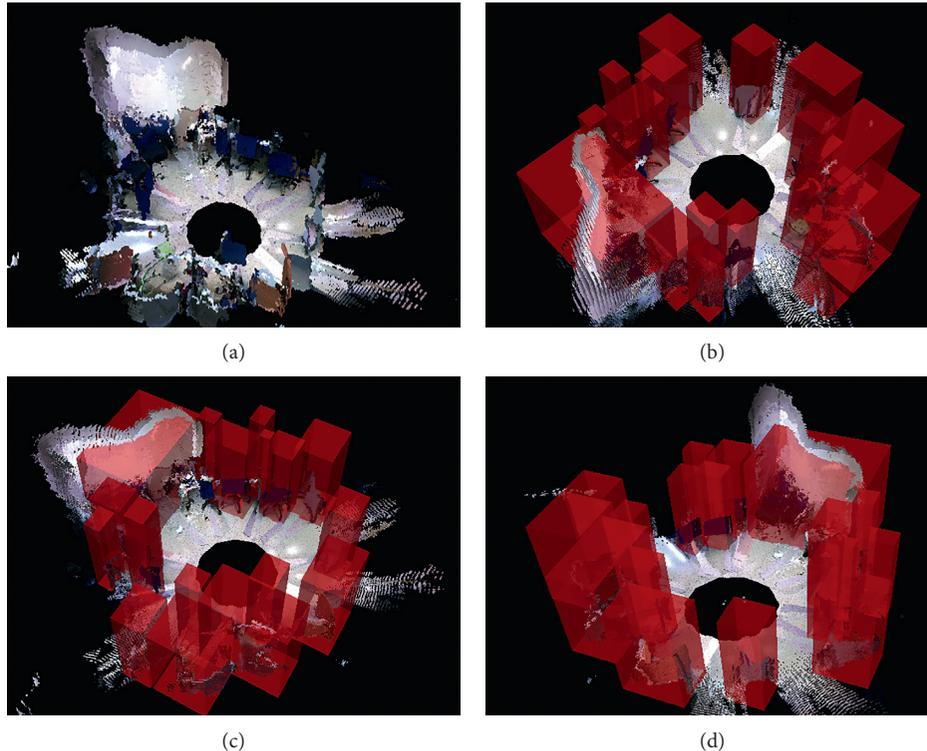


FIGURE 12: Results of detection of the potentially dangerous area in an office environment. (a) The generated map of the environment. (b–d) The results of the OGT algorithm from different points of view.

labeled the bounding boxes of the objects. Finally, we used 60% of the samples for training, 20% were used for validation and the rest for testing. Once we built the dataset, we trained the architecture using the YOLO loss for 25,000 epochs. However, the best intersection over union score was reached at epoch 21,700. This model was selected and involved in the experiments. As a result, the system is able to accurately detect the considered dangerous items and state the area of the image that depict them using an image of the scene capture by the Pepper Robot.

The tridimensional subregion extraction and the centroid computation have no parameters.

We tested the OGT in the same office environment in which we deployed the SOD system. We focused the detection on those that were not considered by the OGT system. We tested the following scenarios:

- (i) Objects lying on the floor that are not considered by the OGT
- (ii) Objects sitting on other objects and obstacles
- (iii) Objects integrated into other objects and obstacles

As mentioned, the OGT failed to detect some objects such as the laptop power adapter and a bottle on the floor. These objects remained unnoticed because certain steps of the OGT algorithms filtered them, but the SOD is able to properly detect them again.

In the OGT experiments, the knife and the power sockets were not detected alone but as part of the table and walls. In this case, the SOD does not contribute with new potentially

dangerous areas but include semantic information on them. However, the semantic information of the object provides highly valuable data to enhance the alerts of the AAL system. For instance, if a falling event is detected by the AAL after the violation of potentially dangerous area of the power socket, the patient may have suffered an electric shock.

The bottle was also ignored by the OGT. The depth sensor cannot compute the distance of plastic surfaces. The only part of the bottle represented in the point clouds is the label. Nonetheless, the amount of points in the label does not exceed the minimum point size of the clustering process of the OGT, so they are filtered. However, the color information correctly depicts the bottle, so the SOD system is able to detect it. Figure 13 shows these experiments.

4.5. Robot Running the Superficial Object Detector Algorithm Experiments in Home Environment. Homes also have multiple sources of potentially dangerous areas such as ovens or electrical panels. Hence, in this experiment, we deployed the robot in an actual home environment and ran the SOD algorithm. We will focus on the same goals as in the last experiment:

- (i) Objects lying on the floor that are not considered by the OGT
- (ii) Objects sitting on other objects and obstacles
- (iii) Objects integrated into other objects and obstacles

In a home, there are not likely to be objects lying on the ground that would be a source of danger. For instance, the

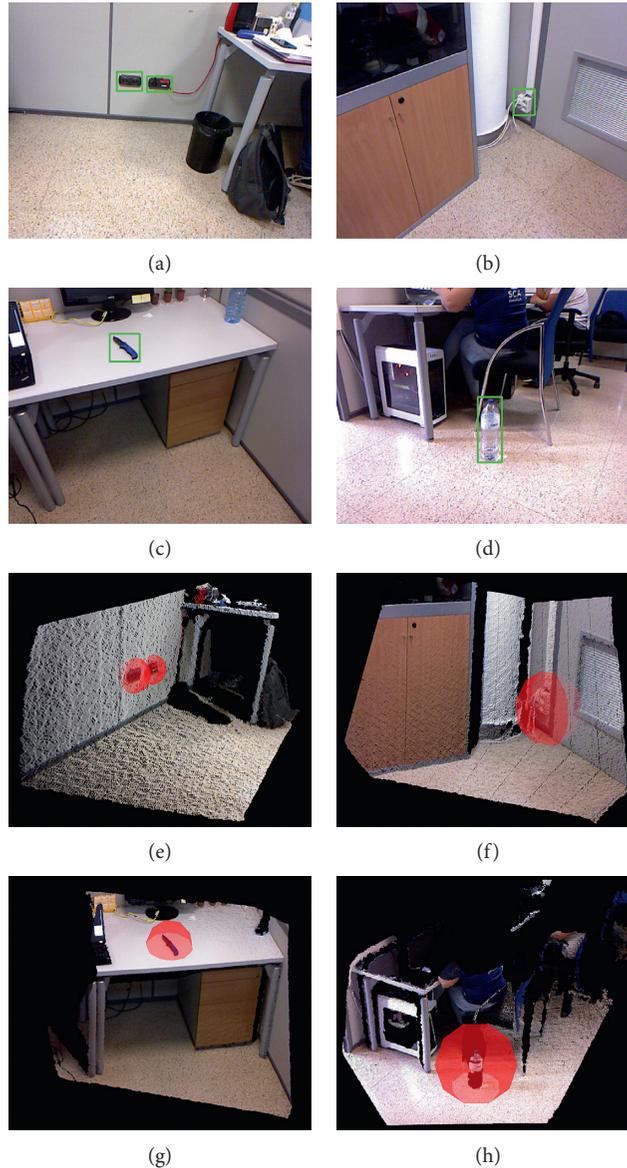


FIGURE 13: Results of detection of the potentially dangerous area on an office environment. (a–d) Color images as captured by the mobile robot. (e–h) Potentially dangerous areas as returned by the SOD algorithm. Spheres are drawn for visualization purposes.

SOD algorithm only detected the family’s dog. Nonetheless, our algorithm also found a variety of potentially dangerous objects that are integrated into, or sitting on, other objects. For instance, the robot vacuum cleaner, the oven, the hob, electrical panels and electrical sockets, and knives were correctly detected as potentially dangerous areas. As mentioned earlier, some of these dangerous areas would be included in the obstacles detected by the OGT so the SOD only contributes the semantic information.

Figure 14 shows some of the potentially dangerous areas detected by the SOD.

4.6. Qualitative Evaluation of a Fall Event. In this experiment, we simulated a fall event in an office environment (Figure 15). When the AAL detected that a person is of the floor (Figures 16(a) and 16(b)), it raises an Alarm using the

common interface between the AAL and the robot (Figure 17). That alarm is composed of the location in which the event was detected, a corresponding image, and the semantic meaning of the event, which in this case is “fall.” If more semantic information is available, it would be included in the alarm. For instance, if the robot previously detected an automatic vacuum cleaner in the room, such information would be included. The alarms are broadcasted to the communication system using a ROS topic. There can be several listeners polling the alarms, for instance, the robot. In this case, the Pepper Robot navigates to the location of the event to check whether it is was a positive detection or not and whether there are additional nonstatic danger sources nearby. The robot also asks the user if he/she needed help. The image from the robot viewpoint, the detected additional sources of danger (if any), and the person’s response to the question of the robot are also included in the alarm.

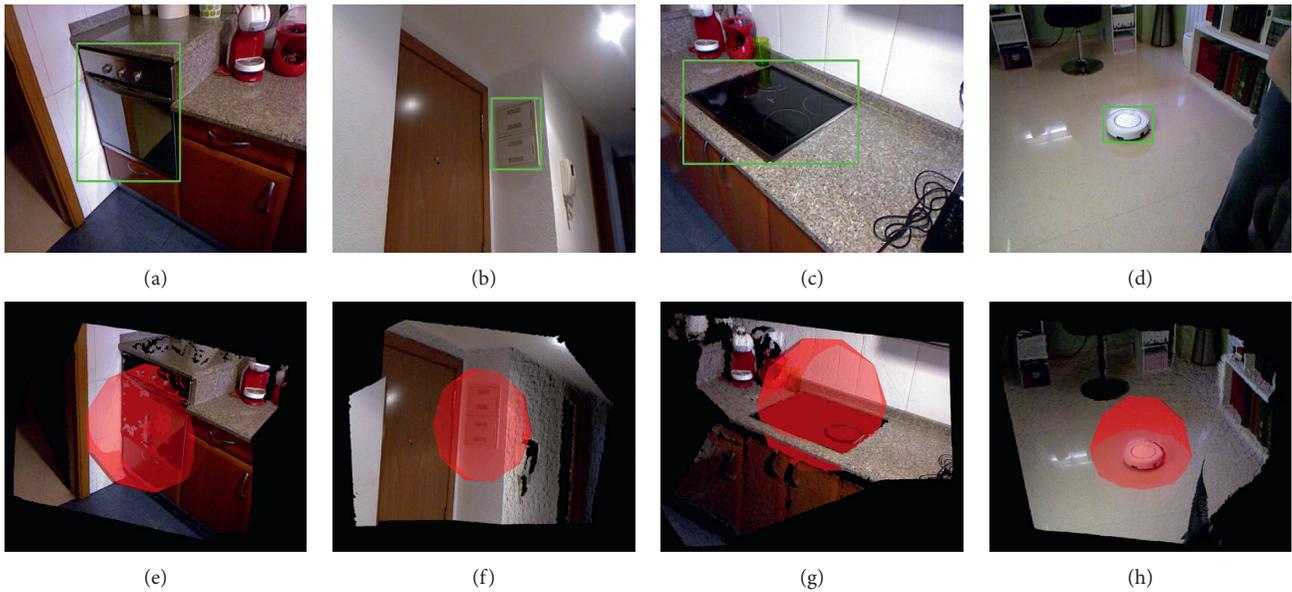


FIGURE 14: Results of detection of potentially dangerous area on a home environment. (a–d) Color images as captured by the mobile robot. (e–h) Potentially dangerous areas as returned by the SOD algorithm. Spheres are drawn for visualization purposes.



FIGURE 15: Risky situation triggered by the AAL with a fall detection.

As mentioned before, several listeners could be polling the alarms besides the robot. For instance, a mobile phone could be also connected so the alarms can reach the person's relatives or an assigned carer.

The response time of each subsystem is reported in Table 3. The AAL system, which is in charge of detecting the person invading a dangerous area and of raising the corresponding alarms, is able to run at about 12 fps. The OGT and SOD systems run in parallel in the external server that is controlling the robot, at about 3 fps. The SLAM algorithms run in the robot's onboard computer at 21 fps.

5. Conclusions

In this paper, the integration of a domestic robot in an ambient assisting living environment is proposed. The AAL system that features an RGB-D camera is able to detect dangerous events such as a person falling or perimeter breaches. Nonetheless, there are small and occluded potentially dangerous areas that cannot be detected with its camera. So, in order to enhance the

AAL capabilities, we propose the utilization of a mobile robot. In this case, a Pepper Robot is in charge of detecting small, nonstatic potentially dangerous areas, such as near wall outlets, a robotic vacuum cleaner, or knives. The position of these objects is forwarded to the AAL, so it can consider more dangerous areas.

We put to test out system in three different environments (home, clinical, and office) with high success. The exhaustive experimentation supports the high accuracy and applicability of the system. In addition, processing times also back the suitability for real-time utilization.

Regarding the limitations of our approach, we realized that the robot tends to accumulate error in the localization process where the features of the location it is at are monotonous. This is due to the way we simulate a laser sensor for the SLAM algorithms. As reported in [38], the precise version of our Pepper Robot has an error-prone depth camera due to a design fault. As a result, the depth data shows high levels of aberrance, thus providing erroneous measures and eventually causing localization errors.



FIGURE 16: Risky situation supervised by the robot.

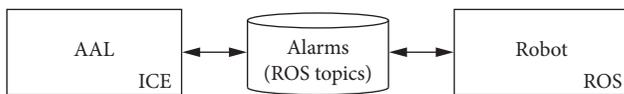


FIGURE 17: The systems that run in the AAL and the methods that are executed in the robot are communicated by using robot operating system (ROS) and ICE.

TABLE 3: Runtime of each subsystem that composes our proposal.

System	Processor	Runtime (ms)
BG subtraction	AAL	56.2
Person detection (3D)	AAL	18.3
Communication overhead	AAL	5.2
Accumulated		79.7
OGT	Robot_DS1	315
SOD	Robot_DS1	25.1
Merging	Robot_DS1	21.5
SLAM	Robot	40
Communication overhead	Robot	7.4
Accumulated		343.9

Note that some processes run simultaneously, so only those that take most time contribute for the accumulated time.

Regarding future research lines, we are exploring the use of convolutional pose machines (CPM) [9, 39, 40] to identify the 3D position of a person’s joints and thus to track their 3D skeleton. CPM techniques have shown good performance in the detection of articulated objects. First, the joints are located in the color images, and second, their 3D positions are estimated taking into account the depth information from RGB-D sensor. This 3D skeleton tracking opens the door to a finer detection of dangerous situations, as arms, head, and legs are estimated separately. In addition, a replacement of the current AAL people detection with a fully deep learning-based approach in real-time is also under development. The robustness provided by neural networks will hopefully improve detection even with static persons and difficult lightning conditions.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Spanish Government TIN2016-76515R Grant, supported with FEDER funds. This work was also supported by a Spanish grant for PhD studies (ACIF/2017/243 and FPU16/00887). Thanks are due to Nvidia for the generous donation of a Titan Xp and a Quadro P6000.

References

- [1] M. Memon, S. Wagner, C. Pedersen, F. Beevi, and F. Hansen, “Ambient assisted living healthcare frameworks, platforms, standards, and quality attributes,” *Sensors*, vol. 14, no. 3, pp. 4312–4341, 2014.
- [2] S. Blackman, C. Matlo, C. Bobrovitskiy et al., “Ambient assisted living technologies for aging well: a scoping review,” *Journal of Intelligent Systems*, vol. 25, no. 1, pp. 55–69, 2016.
- [3] D. Calvaresi, D. Cesarini, P. Sernani, M. Marinoni, A. F. Dragoni, and A. Sturm, “Exploring the ambient assisted living domain: a systematic review,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 8, no. 2, pp. 239–257, 2017.
- [4] N. M. Garcia and J. J. P. C. Rodrigues, *Ambient Assisted Living*, CRC Press, Boca Raton, FL, USA, 2015.
- [5] S. Longhi, P. Siciliano, M. Germani, and A. Monteriù, *Ambient Assisted Living*, Springer, Berlin, Germany, 2014.
- [6] R. Wichert and B. Mand, *Ambient Assisted Living*, Springer, Berlin, Germany, 2017.
- [7] A. Moschetti, L. Fiorini, M. Aquilano, F. Cavallo, and P. Dario, “Preliminary findings of the AALIANCE2 ambient assisted living roadmap,” in *Ambient Assisted Living*, pp. 335–342, Springer, Berlin, Germany, 2014.
- [8] M. Hossain, S. M. R. Islam, F. Ali, K.-S. Kwak, and R. Hasan, “An internet of things-based health prescription assistant and its security system design,” *Future Generation Computer Systems*, vol. 82, pp. 422–439, 2018.
- [9] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in *Proceedings of the 2017 Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.
- [10] P. Cedillo, C. Sanchez, K. Campos, and A. Bermeo, “A systematic literature review on devices and systems for ambient

- assisted living: solutions and trends from different user perspectives,” in *Proceedings of the 2018 International Conference on eDemocracy & eGovernment (ICEDEG)*, pp. 59–66, IEEE, Ambato, Ecuador, April 2018.
- [11] F. Ali, S. M. R. Islam, D. Kwak et al., “Type-2 fuzzy ontology-aided recommendation systems for IoT-based healthcare,” *Computer Communications*, vol. 119, pp. 138–155, 2018.
- [12] M. Ghamari, B. Janko, R. Sherratt, W. Harwin, R. Piechockic, and C. Soltanpur, “A survey on wireless body area networks for ehealthcare systems in residential environments,” *Sensors*, vol. 16, no. 6, p. 831, 2016.
- [13] S. M. Riazul Islam, D. Daehan Kwak, M. Humaun Kabir, M. Hossain, and K.-S. Kyung-Sup Kwak, “The internet of things for health care: a comprehensive survey,” *IEEE Access*, vol. 3, pp. 678–708, 2015.
- [14] S. B. Baker, W. Xiang, and I. Atkinson, “Internet of things for smart healthcare: technologies, challenges, and opportunities,” *IEEE Access*, vol. 5, pp. 26521–26544, 2017.
- [15] E. Clotet, D. Martínez, J. Moreno, M. Tresanchez, and J. Palacín, “Assistant personal robot (APR): conception and application of a tele-operated assisted living robot,” *Sensors*, vol. 16, no. 5, p. 610, 2016.
- [16] S. Payr, F. Werner, and K. Werner, *Potential of Robotics for Ambient Assisted Living*, FFG Benefit, Vienna, Austria, 2015.
- [17] A. Vercelli, I. Rainero, L. Ciferri, M. Boido, and F. Pirri, “Robots in elderly care,” *DigitCult-Scientific Journal on Digital Cultures*, vol. 2, no. 2, pp. 37–50, 2018.
- [18] E. Martínez-Martín and A. P. del Pobil, “Personal robot assistants for elderly care: an overview,” in *Personal Assistants: Emerging Computational Technologies*, pp. 77–91, Springer, Berlin, Germany, 2018.
- [19] N. Ramoly, A. Bouzeghoub, and B. Finance, “A framework for service robots in smart home: an efficient solution for domestic healthcare,” *IRBM*, vol. 39, no. 6, pp. 413–420, 2018.
- [20] D. Fischinger, P. Einramhof, K. Papoutsakis et al., “Hobbit, a care robot supporting independent living at home: first prototype and lessons learned,” *Robotics and Autonomous Systems*, vol. 75, pp. 60–78, 2016.
- [21] H.-M. Gross, S. Mueller, C. Schroeter et al., “Robot companion for domestic health assistance: implementation, test and case study under everyday conditions in private apartments,” in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5992–5999, IEEE, Hamburg, Germany, September–October 2015.
- [22] A. Di Nuovo, F. Broz, N. Wang et al., “The multi-modal interface of robot-era multi-robot services tailored for the elderly,” *Intelligent Service Robotics*, vol. 11, no. 1, pp. 109–126, 2018.
- [23] A. Cesta, G. Cortellesa, A. Orlandini, and L. Tiberio, “Long-term evaluation of a telepresence robot for the elderly: methodology and ecological case study,” *International Journal of Social Robotics*, vol. 8, no. 3, pp. 421–441, 2016.
- [24] S. Koceski and N. Koceska, “Evaluation of an assistive telepresence robot for elderly healthcare,” *Journal of Medical Systems*, vol. 40, no. 5, p. 121, 2016.
- [25] K. Wada and T. Shibata, “Living with seal robots-its socio-psychological and physiological influences on the elderly at a care house,” *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 972–980, 2007.
- [26] G. Brady, S. Roy, and W. George, “Mobile robots and autonomous ambient assisted living,” *Paladyn, Journal of Behavioral Robotics*, vol. 6, no. 1, 2015.
- [27] R. Esposito, L. Fiorini, R. Limosani et al., “Supporting active and healthy aging with advanced robotics integrated in smart environment,” in *Optimizing Assistive Technologies for Aging Populations*, pp. 46–77, IGI Global, Pennsylvania, PA, USA, 2016.
- [28] N. Hendrich, H. Bistry, and J. Zhang, “Architecture and software design for a service robot in an elderly-care scenario,” *Engineering*, vol. 1, no. 1, pp. 27–35, 2015.
- [29] M. M. A. de Graaf, S. B. Allouch, and J. A. G. M. van Dijk, “Long-term evaluation of a social robot in real homes,” *Interaction Studies*, vol. 17, no. 3, pp. 462–491, 2016.
- [30] M. M. A. de Graaf, S. B. Allouch, and J. A. G. M. van Dijk, “Why would i use this in my home? a model of domestic social robot acceptance,” *Human-Computer Interaction*, vol. 34, no. 2, pp. 115–173, 2019.
- [31] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [32] S. Li, C. Xu, and M. Xie, “A robust O(n) solution to the perspective-n-point problem,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1444–1450, 2012.
- [33] H. Arthur and D. Kartashov, “A slam research framework for ros,” in *Proceedings of the 12th Central and Eastern European Software Engineering Conference in Russia (CEE-SECR’16)*, pp. 12:1–12:6, ACM, Moscow, Russia, October 2016.
- [34] M. Quigley, K. Conley, B. P. Gerkey et al., “Ros: an open-source robot operating system,” in *Proceedings of the ICRA Workshop on Open Source Software*, Kobe, Japan, May 2009.
- [35] S. Chakraborty and J. M. Canas, “Making compatible two robotic middlewares: ros and jderobot,” in *Proceedings of the XVII Workshop on Physical Agents (WAF-2016)*, pp. 147–154, Málaga, Spain, June 2016.
- [36] Internet Communications Engine, <https://github.com/zeroc-ice/ice>.
- [37] R. Joseph and F. Ali, “YOLOv3: an incremental improvement,” CoRR, 2018, <http://arxiv.org/abs/1804.02767>.
- [38] Z. Bauer, F. Escalona, E. Cruz, M. Cazorla, and F. Gomez-Donoso, “Improving the 3D perception of the pepper robot using depth prediction from monocular frames,” R. F. Pizán, A. G. Olaya, M. P. S. Lorente, J. A. I. Martínez, and A. L. Espino, Eds., in *Proceedings of the 19th International Workshop of Physical Agents (WAF 2018)*, pp. 132–146, Springer International Publishing, Madrid, Spain, November 2019.
- [39] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, “Hand keypoint detection in single images using multiview bootstrapping,” in *Proceedings of the 2017 Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.
- [40] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” in *Proceedings of the 2016 Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016.

Research Article

Distinguishing Different Emotions Evoked by Music via Electroencephalographic Signals

Yimin Hou ¹ and Shuaiqi Chen ²

¹*School of Automation Engineering, Northeast Electric Power University, Jilin, China*

²*Luneng New Energy (Group) Co., Beijing, China*

Correspondence should be addressed to Yimin Hou; [ymh7821@163.com](mailto:yhm7821@163.com)

Received 1 October 2018; Revised 25 December 2018; Accepted 28 January 2019; Published 6 March 2019

Guest Editor: Anastassia Angelopoulou

Copyright © 2019 Yimin Hou and Shuaiqi Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Music can evoke a variety of emotions, which may be manifested by distinct signals on the electroencephalogram (EEG). Many previous studies have examined the associations between specific aspects of music, including the subjective emotions aroused, and EEG signal features. However, no study has comprehensively examined music-related EEG features and selected those with the strongest potential for discriminating emotions. So, this paper conducted a series of experiments to identify the most influential EEG features induced by music evoking different emotions (calm, joy, sad, and angry). We extracted 27-dimensional features from each of 12 electrode positions then used correlation-based feature selection method to identify the feature set most strongly related to the original features but with lowest redundancy. Several classifiers, including Support Vector Machine (SVM), C4.5, LDA, and BPNN, were then used to test the recognition accuracy of the original and selected feature sets. Finally, results are analyzed in detail and the relationships between selected feature set and human emotions are shown clearly. Through the classification results of 10 random examinations, it could be concluded that the selected feature sets of Pz are more effective than other features when using as the key feature set to classify human emotion statuses.

1. Introduction

Recognition of emotion state is an important aim for the development of advanced brain-computer interfaces (BCIs). For this application, emotion recognition using the electroencephalogram (EEG) has garnered widespread interest due to the convenience, high resolution, and reliability of EEG recordings. Music can evoke powerful emotions, and these emotions are associated with distinct EEG signal patterns. Identification of the EEG signals associated with these emotions may help elucidate the neurological basis for music appreciation, contribute to the development of music programmes for mood therapy [1], and provide biomarkers and novel methods for studying neuropsychiatric disorders such as depression and Alzheimer's disease [2].

Numerous studies have identified EEG signals associated with distinct features of music, including familiarity, level of processing, phrase rhythm, and subjective

emotional response. Thammasan et al. extracted power density spectra and fractal dimensions from the Database for Emotion Analysis using Physiological Signals (DEAPs) and found that using low familiarity music improved the accuracy of recognition regardless of whether the classifier was support vector machine (SVM), multilayer perception, or C4.5 [3]. Kumagai et al. investigated the relationship between cortical response and familiarity of music. They found that the two peaks of the cross-correlation values were significantly larger when listening to unfamiliar or scrambled music compared with familiar music. Collectively, these findings suggest that the cortical response to unfamiliar music is stronger than that to familiar music and so is more appreciate for classification applications by BCIs [4]. Santosa et al. design series of experiments, in which, different level noise, such as no noise (NN), midlevel noise (MN), and high-level noise (HN), was added into the music. The 14 subjects were tested in four different auditory

environments: music segments only, noise segments only, music + noise segments, and the entire music interfered by noise segments. Then, their responses data were analyzed to determine the effects of background noise on the hemispheric lateralization in music processing [5]. Hong and Santosa investigated whether activations in the auditory cortex caused by different sounds can be distinguished using functional near-infrared spectroscopy (fNIRS) [6]. Bigliassi et al. used musical stimuli as interference to examine how the brain controls action when processing two tasks at the same time [7]. Lopata et al. reported stronger activity of the frontal lobe alpha band, which is correlated with the mental imagery of music after it has been played and studied, in subjects with music improvisation training compared with subjects without training. Thus, level of processing (e.g., creative processing) influences the EEG response. Moreover, these results suggest that musical creativity can be learned and improved and that these changes are measurable by EEG [8]. Phneah and Nisar compared the EEG responses induced by favourite subject-selected music to “relaxing” music selected based on alpha wave binaural beats and found that the relaxing music had stronger and longer-lasting physiological and psychological soothing effects [9]. Thus, EEG signals can help in the selection of mood-modifying music. Bridwell et al. compared the EEG responses evoked by structured guitar notes and random notes and found a waveform at 200 ms that was more strongly activated by the structured sequence. Further, the result of this study is that 4 Hz note patterns appears somewhat distinct from the sensitivity of statistical regularities of “auditory oddball” [10]. Martínez-Rodrigo et al. found distinct EEG responses in theta and alpha bands to phrase rhythm variations of two classical sonatas, one in bipartite form and the other in rondo form [11]. Zhou et al. compared the processing of musical meaning conveyed by direction of pitch change in congenital amusia. Twelve Mandarin-speaking amusia and 12 controls performed a recognition (implicit) and a semantic congruency judgement (explicit) task while their EEG waveforms were recorded. The authors concluded that amusia are able to process iconic musical meaning through multiple acoustic cues in natural musical excerpts but not through the direction of pitch change [12]. Many researchers studied the relationships between human emotions and their EEG signal. Lu et al. selected nine musical passages as stimuli and divided them into three groups using variance test and *t*-test according to the two-dimensional model of emotion. To analyse EEG signals, they extracted the power density spectra of different bands and used principle component analysis (PCA) dimensionality reduction for feature selection. They found that emotion recognition accuracy by SVM was higher using the average power information of beta and gamma bands compared with other bands. Thus, beta and gamma bands appear to contain signal information useful for emotional discrimination [13]. Di et al. presented an analysis procedure in order to study the affect of human emotion using EEG characteristics induced by sound stimuli with different frequencies [14]. Kurbalija et al. proposed a method of emotion recognition from EEG

signals using distance-based time-series classification. The results showed that the EEG signal could be used successfully to construct models for recognition of emotions, individuals, and other related tasks [15]. Kobayashi and Nakagawa presented the emotion fractal analysis method (EFAM) in their paper. They assessed emotions based on EEG data to propose a BCI system which can recognize emotions including delight, anger, sorrow, and pleasure and used that information to manipulate an electric wheelchair [16]. Zhao et al. sampled the EEG signal of the volunteers when they watched affective films. After extracting the EEG features, SVM was employed as the classifier to recognize human emotions [17]. Ivanovic et al. collected EEG data from ten males between 25 and 40 years old and presented an automatic real-time classification algorithm using EEG data cited by self-induced emotions [18]. Yoon and Chung proposed a Bayes’ theorem-based classifier which used supervised learning algorithm to recognize human emotion according to the volunteers’ EEG data. In the recognition, Fast Fourier Transform (FFT) analysis was used for feature extraction [19].

The complexity of EEG signals has hampered the selection of optimal feature sets for discrimination of emotions evoked by music. To compensate for the lack of single feature recognition, many previous studies have attempted to extract one or several linear or nonlinear dynamic characteristics of EEG signals for distinguish different music stimuli by machine learning. For example, to study and prevent mild depression, Li et al. extracted 816 features (17 features of 3 frequency bands from 16 electrodes) and improved the recognition rate using CFS dimension reduction and machine learning. They also suggested that signals from electrode positions FP1, FP2, F3, O2, and T3 are most strongly related to mild depression [20]. Xu and Zuo proposed an algorithm based on mutual information and PCA to compensate for the lack of nonlinear relationships between the features of PCA alone using the database of the 2005 international BCI competition. After joint power spectrum estimation, continuous wavelet analysis, wavelet package analysis, and Hjorth parameter calculations, they used the newly proposed algorithm for selection and compared results to the PCA algorithm alone. They found that dimension reduction by the proposed algorithm improved classification accuracy using SCM as the classifier [21].

Though many achievements have been gained in this field, there are still some problems for emotion recognition. Firstly, the main results in emotional recognition focus on visual emotion recognition. The researches in recognition of emotion evoked by music are less and the recognition rates are lower. But in human daily life, music-induced emotions are effective and persistent, allowing a clearer observation of brain activity in this emotional state [22]. Secondly, the spatial resolution of EEG signal is low. So, the effect of the study of brain cognitive rules based on features is poor; most of them can only be precise to a certain brain area, but it is impossible to put forward more precise brain cognitive mechanism. The effectiveness of the study of brain cognitive rules based on features is unsatisfactory.

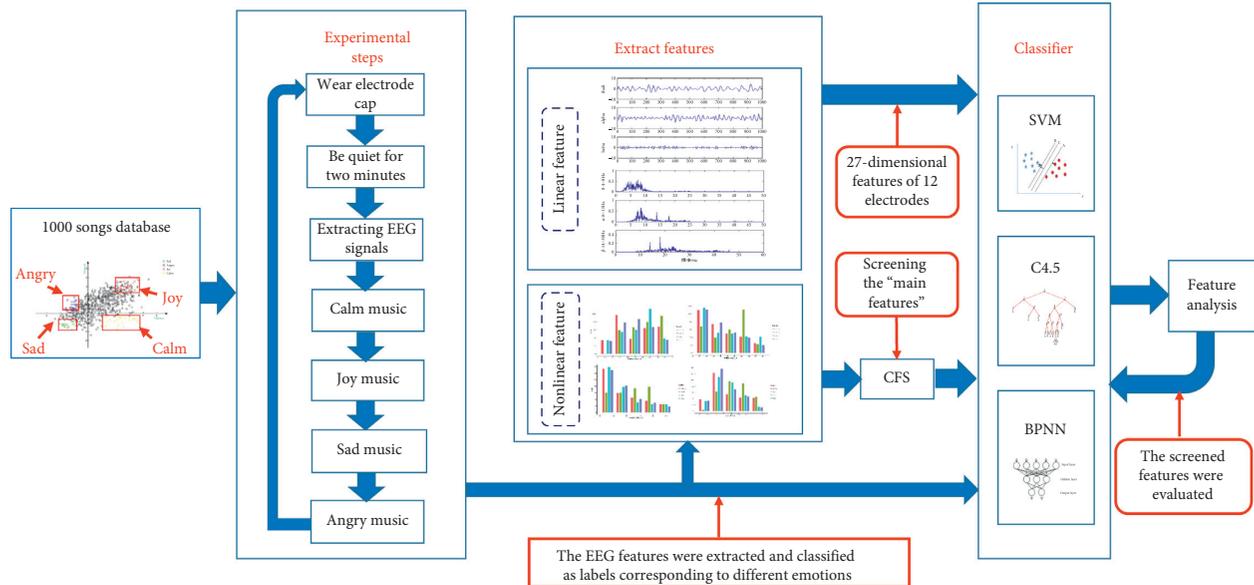


FIGURE 1: Study flow chart.

Most of them can only be located to a certain brain area, but it is difficult to construct the relationships between EEG electrode and emotions' classification [23]. Lastly, due to the complexity of the EEG, most previous studies on the relationships between EEG signal features and music-evoked emotions have focused on the analysis of a few specific characteristics. However, few studies have conducted a comprehensive unbiased analysis of whole-brain EEG signals associated with music-evoked emotions and then selected those with highest discriminative power for various classifiers [24].

In the presented study, we want to obtain the most influential EEG signal feature set of the human emotion classification. To achieve this goal, 18-dimension linear features and 9-dimension nonlinear features were extracted for every electrode, and the correlation-based feature selection (CFS) method was employed to select the influential feature set. To verify the influence of the selected feature set, the classification methods including BPNN, SVM, C4.5, and LDA were used in the procedure of human emotion classification. The experiment results showed that the selected feature set of Pz electrode and the classification method C4.5 were more effective in human emotion recognition.

2. Methods

As shown in Figure 1, there were five stages in this study: collection of volunteers' subject data, EEG recording during listening music, extraction of EEG features, emotion classification, and detailed analysis for validation, dimension reduction, and accuracy improvement.

2.1. Source of Music Stimuli. The music stimuli used in this study are from the database of 1000 songs, which was selected by Mohammad Soleymani and colleagues of the University of Geneva for emotion analysis from the Free

Music Archive (<http://freemusicarchive.org/>). For each song, the sampling rate in the database is 44,100 Hz and the length is 45 s. Each song is also marked with the arousal dimension and the valence dimension for classification by the two-dimensional model [25, 26]. For classification, we separated the data into four groups according to the highest average scores and variance and there were 22 samples in "joy" and "calm" group, while 20 samples in "sad" and "angry" group. To balance the data number, 20 music samples were chosen for every emotion. The average scores, variance of arousal, and valence values were provided by the above database. As shown in Figure 2, there was no significant difference between the scores in different groups by checking the correlation between the groups using analysis of variance (ANOVA). As shown in Table 1, scores in arousal dimension of sad and calm are same; scores in valence dimension of angry and calm are same. So, we just compared 12 pairs in the table which showed the homogeneity test of variance. Then, according to the reviewer's suggestion, we used the analysis of variance in Table 2. It is showed that the P value is all less than 0.05, indicating that there are significant differences in the four emotional statuses in valence and arousal dimension. Four kinds of emotions can be regarded as different kinds of data and used as different stimulus to cause EEG signals of audiences in experiments [27]. In the experiment, the data with highest scores for both arousal and valence dimension were defined as "happy" or "joy," that with lowest scores for the two dimensions were defined as "sad," and that with highest arousal and lowest valence as "angry" and defined the data as "calm" in the opposite case.

2.2. EEG Recording. The participants were eight graduate students who have not music background (23.11 ± 3.14 years old, six males and two females). Before the experiment, the subjects provided personal information and informed

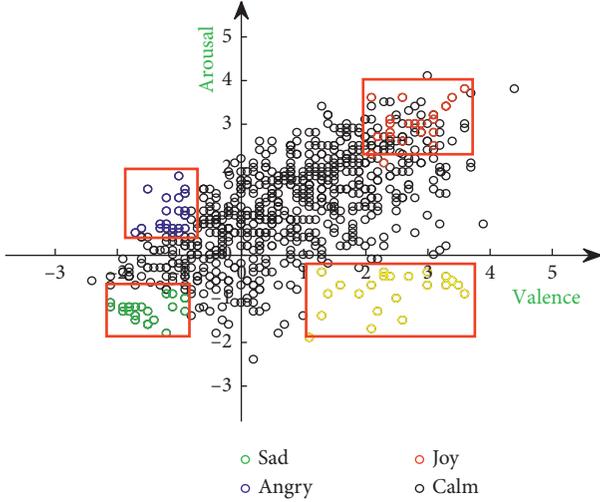


FIGURE 2: Emotional classification of the sample music.

TABLE 1: Homogeneity test of variance.

Pairs	Levene statistic	df1	df2	Saliency
sad_arousal-clam_arousal	2.445 ^a	7	12	0.083
sad_arousal-joy_arousal	2.905 ^b	7	12	0.050
sad_arousal-angry_arousal	2.749 ^c	7	12	0.054
clam_arousal-joy_arousal	1.036 ^a	4	14	0.423
clam_arousal-angry_arousal	1.392 ^b	4	14	0.287
joy_arousal-angry_arousal	0.473 ^a	4	14	0.755
sad_valence-clam_valence	0.398 ^a	5	13	0.842
sad_valence-joy_valence	2.512 ^b	5	13	0.084
sad_valence-angry_valence	2.307 ^c	5	13	0.104
clam_valence-joy_valence	1.219 ^a	6	13	0.357
clam_valence-angry_valence	0.969 ^b	6	13	0.483
joy_valence-angry_valence	2.516 ^a	3	11	0.058

a, b, and c are the ignored abnormal values

consent. They were requested to avoid stimuli and refrain from activities that may induce strong emotions before one day of the experiment. At the beginning of the experiment, EEG electrodes were applied, and subjects were requested to remain calm with eyes closed for 2 min without playing any music. To avoid carryover effects, the music was played in the order calm, happy, sad and angry during EEG recordings. Each music passage was 45 s long, and to get same music number, we select 20 music passages in each music emotion group. To avoid volunteer fatigue, subjects rested for approximately 10–15 min between each block of 20 music passages in a given group. Every subject was asked to repeat the experiment 2 or 3 days later, yielding a total of 16 datasets. Due to failure of one trial, however, only 15 datasets were obtained.

A NCC MEDICAL NCERP-P series EEG system and 24 electrodes EEG map was used for all recordings (Figure 3), and EEG electrodes were arranged according to the 10–20 international system. The average resistance of bilateral papillae reference and scalp recording electrodes was 5 k-Ohm, the sampling rate was 256 Hz, and the power frequency was 50 Hz. Before data processing, the eye power was filtered by ICA on EEG devices. The original signal is

filtered through adaptive filtering to remove 50 Hz power frequency noise, and the final signal is obtained through wavelet filtering and preprocessing.

3. Data Analysis

3.1. Feature Extraction. Many previous studies on EEG correlates of human emotion have examined only linear features [28–30]. However, developments in nonlinear dynamics have enhanced our understanding of the brain as a high-dimensional complex chaotic system, and nonlinear dynamic characteristics are now widely used in EEG research [31–33]. As shown in Figure 4, the EEG data acquired were first preprocessed and filtered for noise and the 50 Hz frequency signal filtered by adaptive filtering. The ICA signal was then filtered out to extract a 15 s EEG epoch for each music passage which is located in the middle of the 45 s music. This is because we want to avoid volunteers' mood swing at the beginning of the testing and fatigue at the end of the testing. So, the first and the last 15 s data were removed. We focused on electrodes FP1, FP2, F3, F4, F7, F8, Fz, C3, C4, T3, T4, and Pz as these positions are strongly related to emotion. The 18-dimensional linear features (peak, mean, variance, centre frequency, maximum power, and power sum of 3 frequency bands, theta, alpha, and beta) plus the 9-dimensional nonlinear dynamics features (singular spectral entropy, Lempel–Ziv complexity, spectral entropy, C0 complexity, maximum Lyapunov exponent, sample entropy, approximate entropy, K entropy, and correlation dimension) were extracted.

The linear features were expressed as following. Peak could be presented as

$$Pe = \max[x(n)]. \quad (1)$$

where $x(n)$ is the sampling EEG data.

Mean and variance are just the mean value and variance of $x(n)$. Centre frequency could be expressed as

$$F_a = F_c \times \frac{f_s}{a}, \quad (2)$$

where a is the scale value in wavelet transform and f_s is the sampling frequency; F_c is the wavelet centre frequency when the scale was 1; F_a was the centre frequency when the scale was a . If $r(k)$ was the self-correlation function of EEG data $x(n)$, maximum power could be defined as

$$Pm = \max[P(\omega)], \quad (3)$$

in which,

$$P(\omega) = \sum_{k=-\infty}^{+\infty} r(k)e^{-jak}. \quad (4)$$

Power sum was the summation of $P(\omega)$.

The nonlinear features could be obtained according to the following references. Approximate entropy was presented by Pincus in 1983 [34]. It is a positive number and for EEG signals, the larger the value, the higher the complexity or the stronger the irregularity. Sample Entropy is an improvement of approximate entropy by Richman

TABLE 2: Variance analysis.

Pairs	Sum of squares	df	Mean of square	F	Saliency
sad_arousal * clam_arousal	1.195	7	0.171	5.898	0.033
sad_arousal * joy_arousal	1.377	11	0.125	7.597	0.026
sad_arousal * angry_arousal	2.326	14	0.166	10.661	0.015
clam_arousal * joy_arousal	0.354	11	0.032	521.398	0.001
clam_arousal * angry_arousal	0.654	14	0.047	23.641	0.001
joy_arousal * angry_arousal	2.46	14	0.176	10.649	0.018
sad_valence * clam_valence	0.593	8	0.074	231.87	0.001
sad_valence * joy_valence	1.037	8	0.13	12.58	0.015
sad_valence * angry_valence	0.539	10	0.054	11.042	0.018
clam_valence * joy_valence	1.56	10	0.156	22.03	0.001
clam_valence * angry_valence	0.957	10	0.096	22.464	0.001
joy_valence * angry_valence	2.025	10	0.203	1.16	0.404

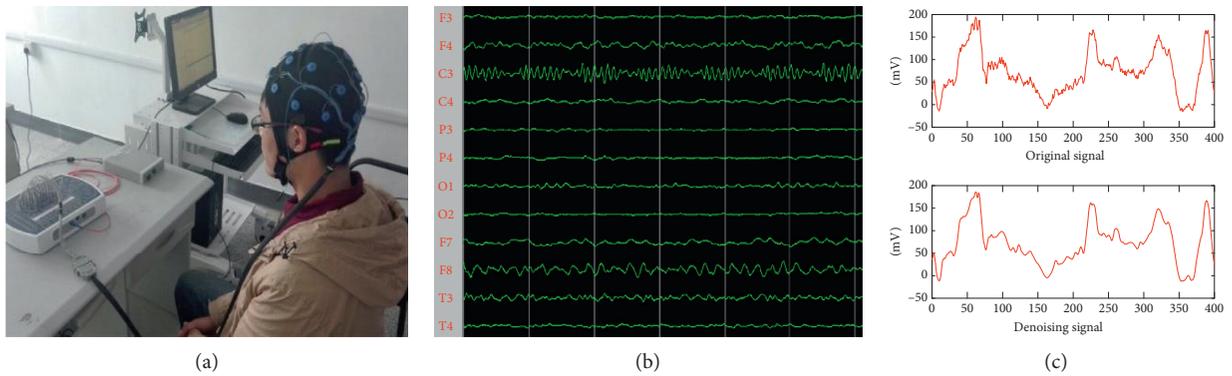


FIGURE 3: EEG acquisition equipment and filtering noise.

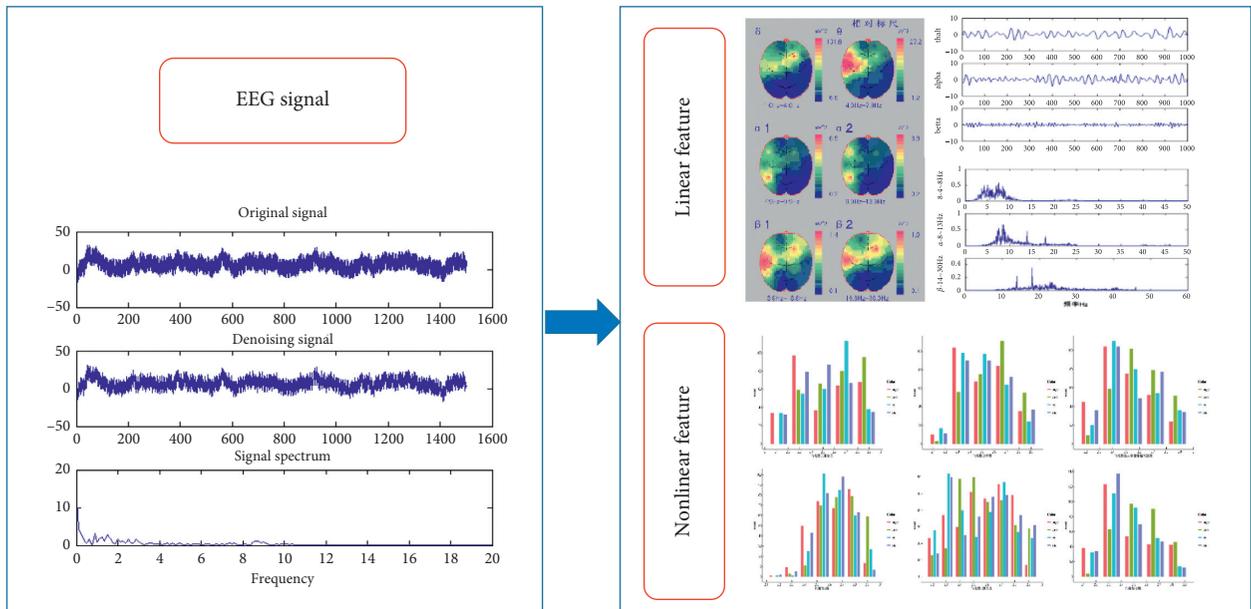


FIGURE 4: Features extracted from the EEG.

and Moorman [35], which can reduce the calculation error and improve the calculation accuracy. Correlation dimension is an important branch of fractal dimension and was proposed by Grassberger and Procaccia in 1983 [36]. K entropy was also called information dimension and was

presented by Kolmogorov and was improved by Sinai. It can be expressed according to reference [37]. The higher the LZ complexity is, the more irregular the change of time series is. It indicates the rate at which a given time series increases with its length to cause a new pattern increasing. The new

pattern shows a decreasing trend with the increase of time, which means that the change of original data series is slower. It can be expressed according to reference [38]. Whether the maximum Lyapunov exponent is greater than zero is the criterion to judge whether the system is chaotic or not, and it can be expressed according to reference [39]. The idea of the C0 complexity is to decompose the time series to be analyzed into random sequence and regular sequence. It can be expressed according to reference [40]. Singular spectrum analysis is to reconstruct the delay of one-dimensional EEG time series into multidimensional phase space. After singular value decomposition, the importance of decomposed quantities is determined according to the order of energy [41]. The spectral entropy can be expressed according to [42]. In the procedure, firstly, the signal is transformed by Fourier transform, then the power distribution of the signal is calculated and the unit power is normalized.

3.2. Feature Selection. The CFS method, which is widely used for feature selection and data cleaning, comprehensively evaluates the correlations between features and classifications as well as the redundancy between features [43, 44]. The core idea of this method is to remove redundant features and select unique class-related features from the original feature set using correlation analysis [45]. Briefly, CFS calculates the correlations between features as well as between features and categories of feature concentration. The calculation formula is shown below:

$$\text{Merit} = \frac{k\overline{r_{cf}}}{\sqrt{k+k(k-1)\overline{r_{ff}}}} \quad (5)$$

Merit is an evaluation of the characteristic subset s , where s contains k characteristics, $\overline{r_{cf}}$ represents the average correlation between feature f ($f \in S$) and category c , and $\overline{r_{ff}}$ represents the average relativity value between features. Formula (5) shows that for the feature subset s , when the correlations between each feature and class label were greater, and the correlations between each two features were smaller, the value of merit would be much greater, and it means that the feature set s was the better one.

The correlation between the features can be calculated using the information gain method. Assuming that y is a possible value of attribute Y , then the entropy of Y is calculated as

$$H(Y) = - \sum_{y \in Y} p(y) \log_2(p(y)). \quad (6)$$

If a certain property X is known, the method to calculate the entropy of Y under X conditions is

$$H(Y | X) = - \sum_{x \in X} p(x) \sum_{y \in Y} p(y | x) \log_2(p(y | x)). \quad (7)$$

The additional information that feature X contributes to Y is called information gain. The correlation between

information gain and two features is positive. Information gain is defined as

$$H(Y) - H(Y | X). \quad (8)$$

Since information gain is a measurement method of symmetry, it must be normalized. To this end, we used the following formula:

$$U_{XY} = 2.0 \times \frac{H(Y) - H(Y | X)}{H(Y) + H(X)}. \quad (9)$$

The correlation coefficient describes the strength of correlation between two variables, with values closer to 1 indicating a stronger correlation.

A greedy progressive search algorithm was used to generate candidate feature subsets from feature sets. Using this algorithm, feature selection produces a feature sequence that is sorted by the degree of correlation $f_1, f_2, f_3, \dots, f_{27}$, then uses the classifier to identify the feature subset $(f_1), (f_1, f_2), (f_1, f_2, f_3), (f_1, f_2, f_3, f_4), (f_1, f_2, f_3, f_4, \dots, f_{25}, f_{26})$.

As shown in Table 3, for every feature, we calculated the average Merit value of every volunteer. From the result, if we set the threshold as 0.1, the average Merit value of feature 4, 8, 10, 14, 16, 17, 20, 21, 22, 23, 25, 26 are higher than the threshold while other features were much lower. So, they were selected as the influential features which were agreed with the data in Table 4.

3.3. Classifier Verification. SVM, decision tree, and neural network are the most common classification methods used in machine learning. In this study, SVM, C4.5, BP neural network, and LDA were used to verify the recognition rate (accuracy) after feature selection. We used the method of 10% cross validation, 10% of data is the training data, then took the mean value of 100 times repetitions to identify the correct rate as the recognition rate. Finally, as shown in Figure 5, for every subfigure, the horizontal axis is the electrodes and the vertical axis is the recognition rate. For every classification method, using Pz, T3, and T4 electrodes' data, we can get much higher recognition rates than using other electrodes. So, these 3 electrodes were chosen in the classification procedure. Moreover, to compare the results in details, the recognition accuracy values using the 3 electrodes' data through the four classification methods are shown in Table 5.

According to classification results, the recognition rates of the features selected by CFS (feature subset) were greater than the recognition rates before dimension reduction using SVM, C4.5, and BP. But for LDA method, the situation was opposite. According to previous validation studies, LDA is a robust emotion classifier [46]. But in the current study, BP, SVM, and C4.5 achieved good classification accuracy.

In order to evaluate the classifier performance, the ROC curves of four different classifiers are shown in Figure 6. The ROC curve of C4.5 and LDA classifier is good, and the AUC average value is larger, so the performance of LDA and C4.5 classifier is better than that of BP and SVM.

For statistical feature analysis, two samples of t -tests

TABLE 3: The differences of average value U_{xy} for every feature.

No.	Feature	U_{xy}
1	Alpha peak	0.042
2	Alpha average value	0.061
3	Alpha variance	0.035
4	Alpha centre frequency	0.146
5	Alpha maximum power	0.024
6	Alpha power sum	0.021
7	Theta peak	0.019
8	Theta average value	0.127
9	Theta variance	0.031
10	Theta centre frequency	0.189
11	Theta maximum power	0.072
12	Theta power sum	0.012
13	Beta peak	0.031
14	Beta average value	0.194
15	Beta variance	0.014
16	Beta centre frequency	0.124
17	Beta maximum power	0.101
18	Beta power sum	0.023
19	Singular spectral entropy	0.055
20	Entropy of K	0.202
21	Approximate entropy	0.138
22	Maximum Lyapunov exponent	0.129
23	Complexity of C0	0.121
24	Sample entropy	0.061
25	Spectral entropy	0.183
26	Lempel–Ziv complexity	0.211
27	Correlation dimension	0.074

TABLE 4: Accuracies for different dimension features subset across different electrode.

No.	Feature	Selected times	Correlation degree
1	Alpha peak	0	Low
2	Alpha average value	2	Low
3	Alpha variance	0	Low
4	Alpha centre frequency	28	High
5	Alpha maximum power	0	Low
6	Alpha power sum	0	Low
7	Theta peak	0	Low
8	Theta average value	41	High
9	Theta variance	0	Low
10	Theta centre frequency	26	High
11	Theta maximum power	7	Low
12	Theta power sum	0	Low
13	Beta peak	0	Low
14	Beta average value	40	High
15	Beta variance	0	Low
16	Beta centre frequency	27	High
17	Beta maximum power	21	High
18	Beta power sum	0	Low
19	Singular spectral entropy	19	Low
20	Entropy of K	30	High
21	Approximate entropy	27	High
22	Maximum Lyapunov exponent	35	High
23	Complexity of C0	31	High
24	Sample entropy	18	Low
25	Spectral entropy	39	High
26	Lempel–Ziv complexity	40	High
27	Correlation dimension	19	Low

were performed on the different emotional features and different electrodes (at significance level $p = 0.05$). Then, each feature of each electrode was tested by paired t -test, and the number of features rejected according to the null hypothesis was recorded. Table 6 examines whether the features of the same electrode are irrelevant when different emotions are stimulated, and the values in the table indicate the number of P values bigger than 0.05. From the t -test results in Table 6, the number of unrelated features was greater for electrodes T4 and Pz than for any other electrode pair.

3.4. Feature Test. The analysis above indicates that the features selected by CFS are more effective for discrimination than the original features. To identify the most effective features in the original feature set, the features in the optimal feature subset, which means the feature set contains n features selected by CFS for one volunteer, were analyzed. Since the optimal feature subset selected by CFS a little differs by volunteer, it is necessary to consider the optimal recognition rate which is get using optimal feature subset, for all volunteers. Therefore, we extracted the recognition rate of all feature subsets for all subjects and verified the best feature subset for most subjects. From the analysis shown in Figure 7, when the feature number n changes from 2 to 27, electrodes T4 and Pz were the most effective for recognition of music-evoked emotions. Moreover, recognition accuracy was better of C4.5 than other classification methods, and the accuracies of T4 and Pz electrodes are much better than that of P3. The data corresponding to Figure 7 are shown in Table 7.

This recognition analysis using signals from different electrodes and different classifiers indicated that recognition accuracy was not improved so much or was even reduced when the selected feature subset had more than 10 dimensions, regardless of the recognition algorithm or EEG channel used. The top 10-dimensional feature subsets of each music group in the 15 datasets were selected to construct a frequency distribution histogram. In the histogram, a more effective feature has higher frequency to be selected by CFS. As shown in Figure 8, for three electrodes, there are 15 volunteers' experiment data of different features in frequency domain. The total chosen times of every feature by CFS for every volunteer are shown in Table 4. We chose the features whose chosen time was higher than 20 as the features correlated with the class labels closely, such as feature 4 (Alpha centre frequency), 8 (Theta average value), 10 (Theta centre frequency), 14 (Beta average value), 16 (Beta centre frequency), 17 (Beta maximum power), 20 (entropy of K), 21 (approximate entropy), 22 (the maximum Lyapunov exponent), 23 (the complexity of C0), 25 (the spectral entropy), and 26 (Lempel–Ziv complexity). The above features included 6-dimensional linear features, and 6-dimensional nonlinear features were the final selected optimal feature subset. This represents the characteristic feature combination most effective for most subjects.

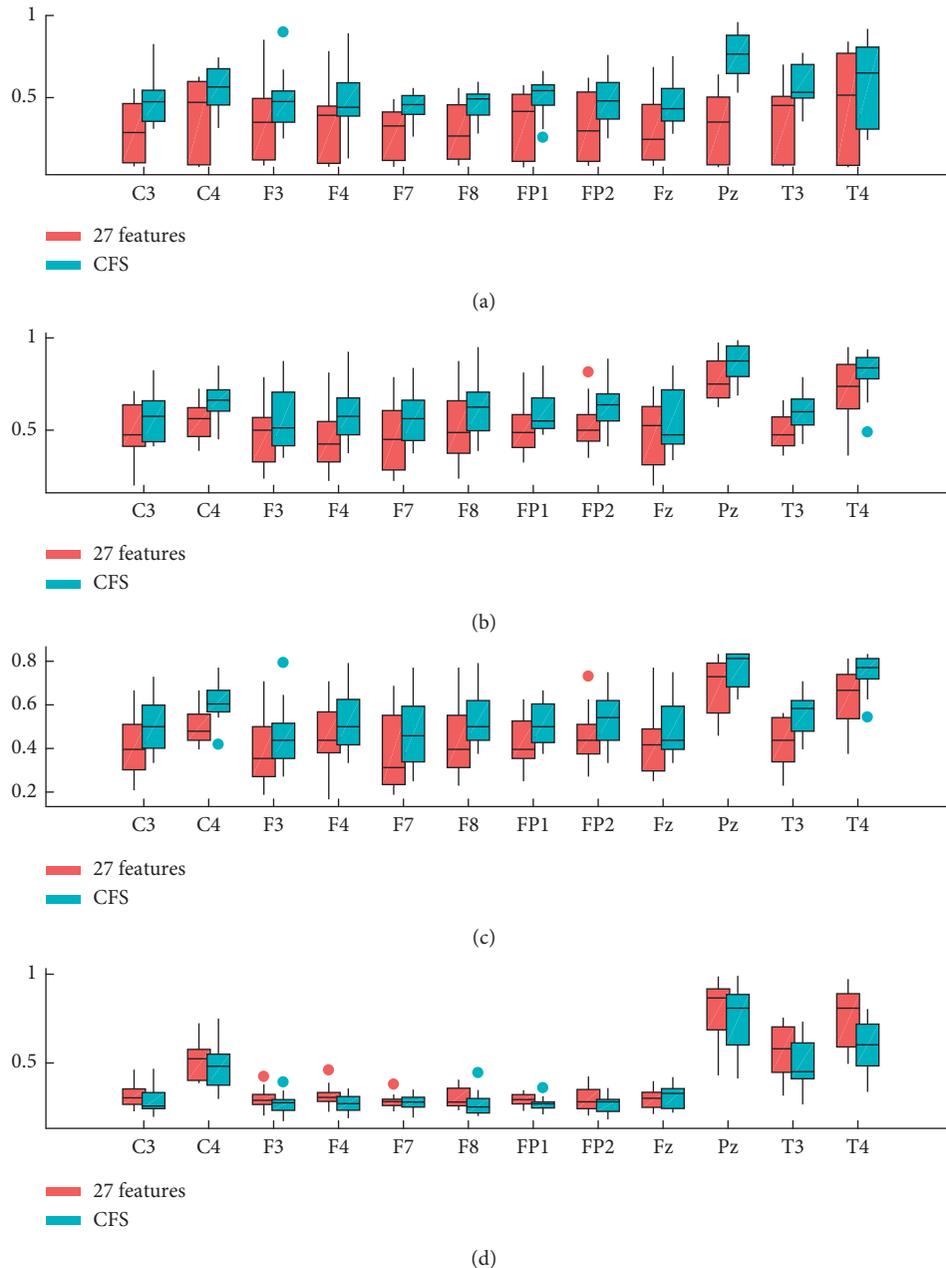


FIGURE 5: The recognition rates of SVM, C4.5, BP, and LDA classifiers before and after dimension reduction. (a) Feature reduction recognition rate verified by SVM. (b) Feature reduction recognition rate verified by C4.5. (c) Feature reduction recognition rate verified by BP. (d) Feature reduction recognition rate verified by LDA.

4. Experiments and Analysis

4.1. Linear Features. We found that the beta wave accounted for the largest proportion of linear features selected by CFS, consistent with previous research on emotion recognition [47]. Of the linear features, the band centre frequency was also of obvious importance, possibly because the centre frequency best represents and distinguishes the band.

In this study, EEGLAB was used to analyse the linear features. The steps are as follows. First, the wavelet was used to process the frequency-band divisions (theta, alpha, and beta) of the 15 datasets. Then, the EEG data acquired during music

evoking the four emotions, and the associated frequency bands were superimposed on the average. Finally, EEGLAB was imported and brain topographic maps constructed for each emotion using the 12 emotion-related electrodes.

As shown in Figure 9, when the subject was listening to angry or quiet music, band energy was higher in the frontal area. When listening to joyful music, theta and alpha band energy was higher in the occipital cortex region, while beta band energy was higher at the forehead. When listening to sad music, the alpha wave exhibited a wider activity range. Comparing these activity patterns, the alpha band appears to change mostly with emotion evoked by music, suggesting

TABLE 5: The classification accuracy through the 4 methods using 27 features and the CFS feature set.

Methods	Feature set	Electrode	Recognition rate (%)
SVM	27 original features	Pz	47.68
		T3	51.22
		T4	57.49
	Feature set selected by CFS	Pz	75.42
		T3	55.83
		T4	63.74
C4.5	27 original features	Pz	72.24
		T3	52.56
		T4	73.32
	Feature set selected by CFS	Pz	85.46
		T3	63.78
		T4	80.95
BP	27 original features	Pz	79.63
		T3	54.64
		T4	72.27
	Feature set selected by CFS	Pz	82.55
		T3	62.96
		T4	78.71
LDA	27 original features	Pz	92.23
		T3	63.54
		T4	88.65
	Feature set selected by CFS	Pz	89.78
		T3	56.89
		T4	70.43

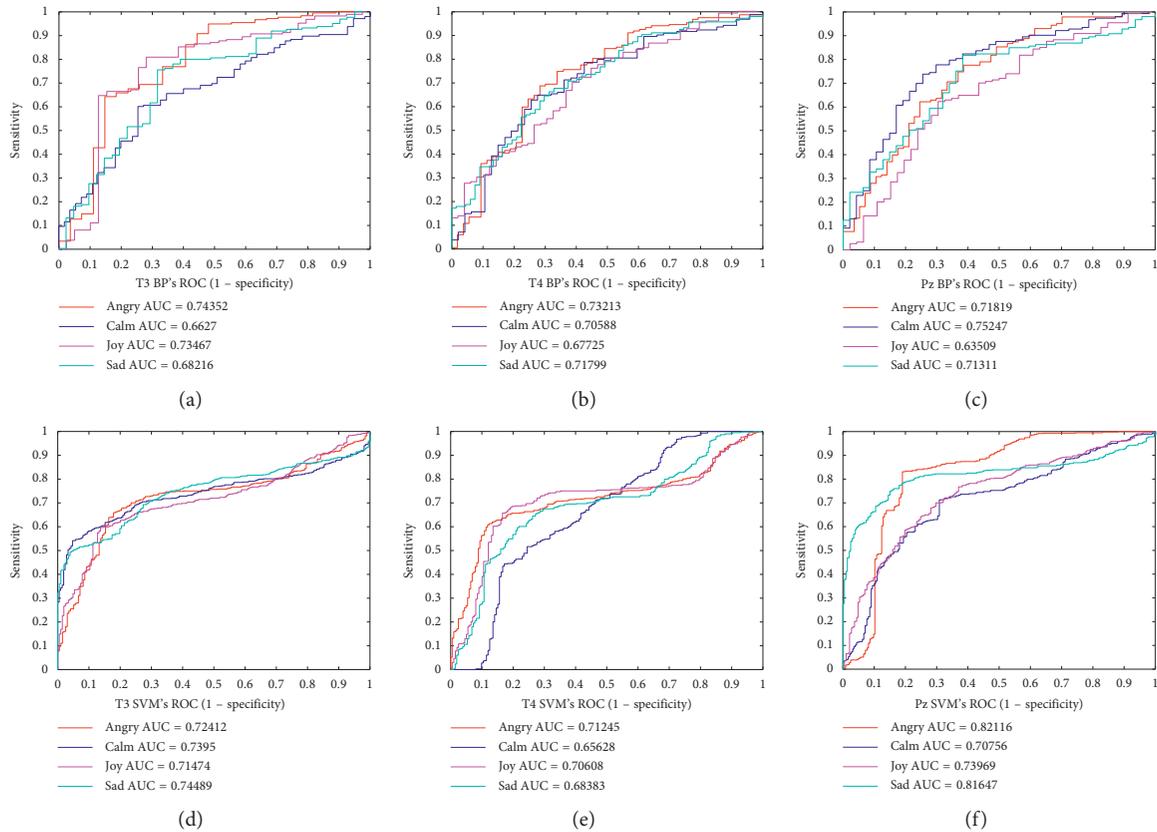


FIGURE 6: Continued.

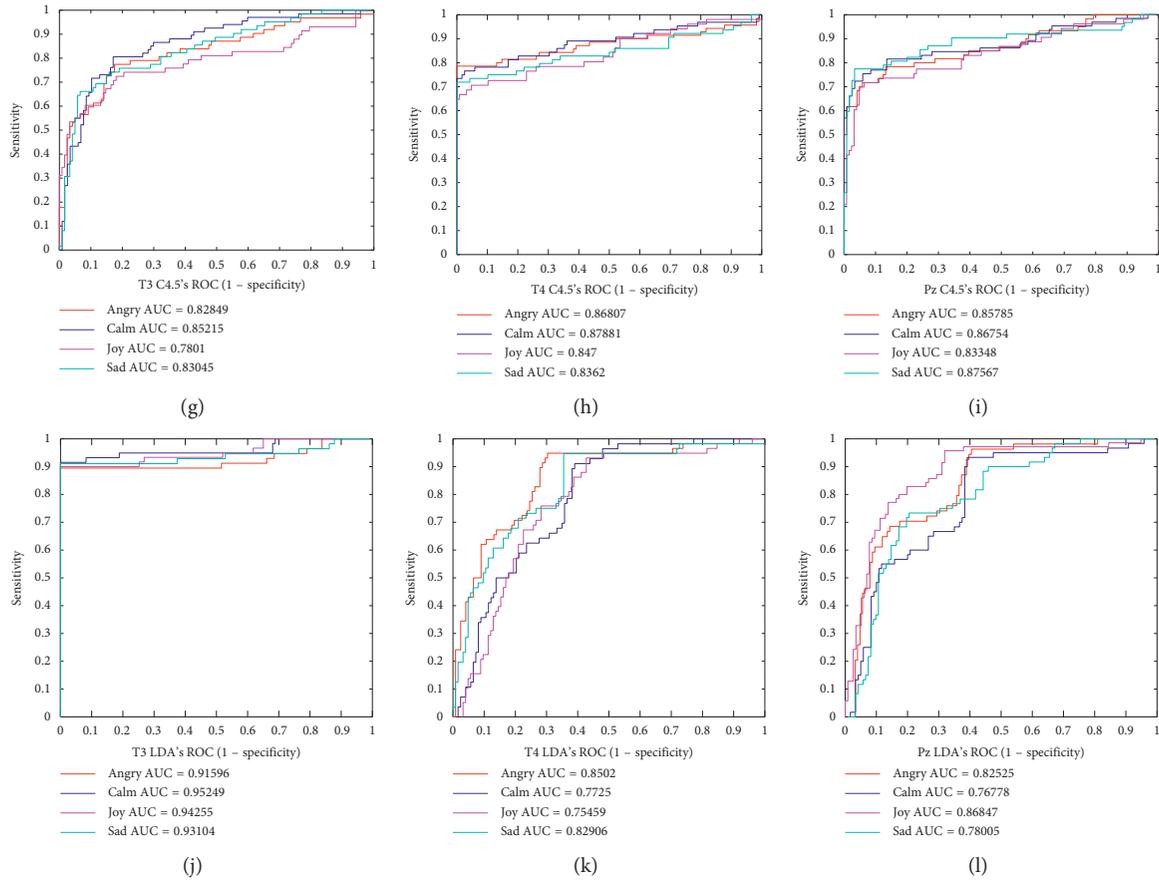


FIGURE 6: The ROC curves of SVM, C4.5, BP, and LDA classifiers.

TABLE 6: The unrelated numbers of 12 electrodes of different emotions using paired t -test ($p > 0.05$).

Electrode	FP1	FP2	F3	F4	F7	F8	Fz	C3	C4	T3	T4	Pz
Anger-calm	146	135	139	144	136	112	138	130	125	94	168	152
Anger-joy	132	116	107	130	109	105	115	126	76	92	157	142
Anger-sad	99	126	83	99	100	97	100	117	66	67	120	127
Calm-joy	126	124	101	116	97	119	121	109	99	80	124	129
Calm-sad	161	151	127	138	151	133	137	149	126	107	179	164
Joy-sad	112	127	94	111	116	110	103	127	89	103	162	145
Total	776	779	651	738	709	676	714	758	581	543	910	859

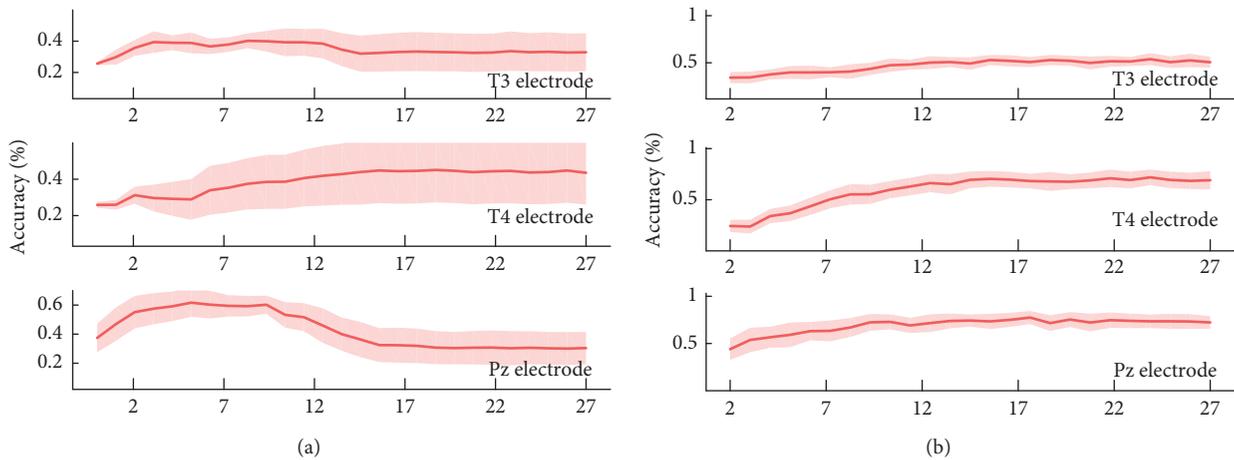


FIGURE 7: Continued.

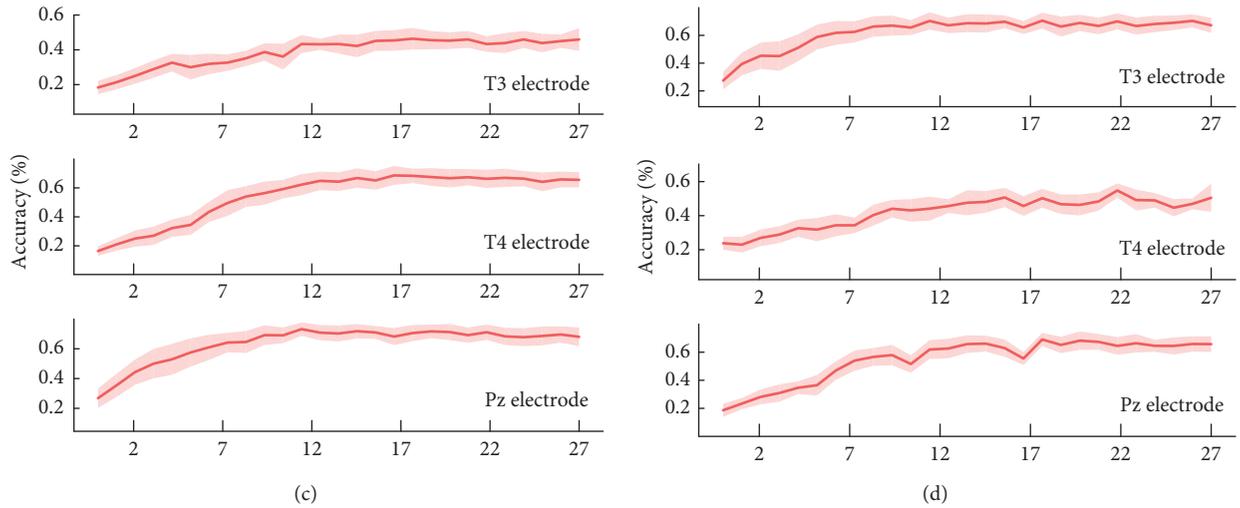


FIGURE 7: Recognition accuracies of different feature subset from electrodes T3, T4, and Pz verified by different classifiers. (a) Recognition accuracy of different feature subsets of T3, T4, and Pz verified by SVM. (b) Recognition accuracy of different feature subset of T3, T4, and Pz verified by C4.5. (c) Recognition accuracy of different feature subset of T3, T4, and Pz verified by BP. (d) Recognition accuracy of different feature subset of T3, T4, and Pz verified by LDA.

TABLE 7: Accuracies for different dimension features subset across different electrode.

Feature number	2	5	10	15	20	25
SVM (%)						
T3	25.35	40.96	42.95	37.63	38.64	38.99
T4	24.75	30.67	41.66	42.86	43.52	46.23
Pz	40.67	61.53	60.97	41.63	39.94	38.01
C4.5 (%)						
T3	37.56	40.82	50.67	56.32	55.77	56.18
T4	36.64	57.63	73.05	74.57	74.69	76.77
Pz	57.81	67.74	75.19	78.63	77.88	78.90
BP (%)						
T3	20.23	32.74	40.12	45.64	47.57	48.92
T4	19.27	38.69	62.45	63.74	62.29	64.67
Pz	27.54	58.78	73.13	71.62	70.87	71.33
LDA (%)						
T3	25.63	59.67	64.78	65.14	64.97	65.34
T4	21.78	32.71	45.91	50.83	47.63	46.92
Pz	20.79	38.98	58.67	62.38	63.00	62.47

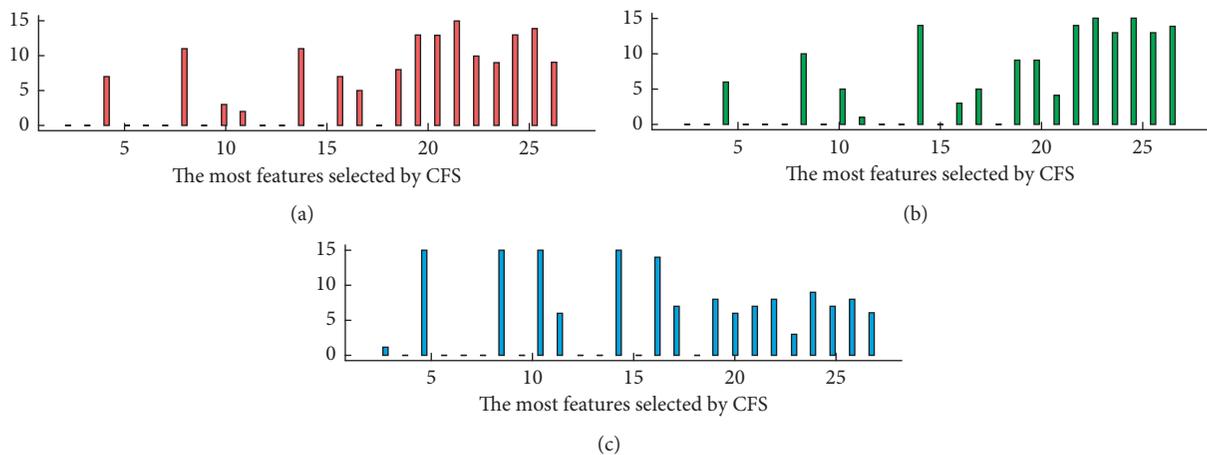


FIGURE 8: Frequency histograms of features from electrodes T3, T4, and Pz.

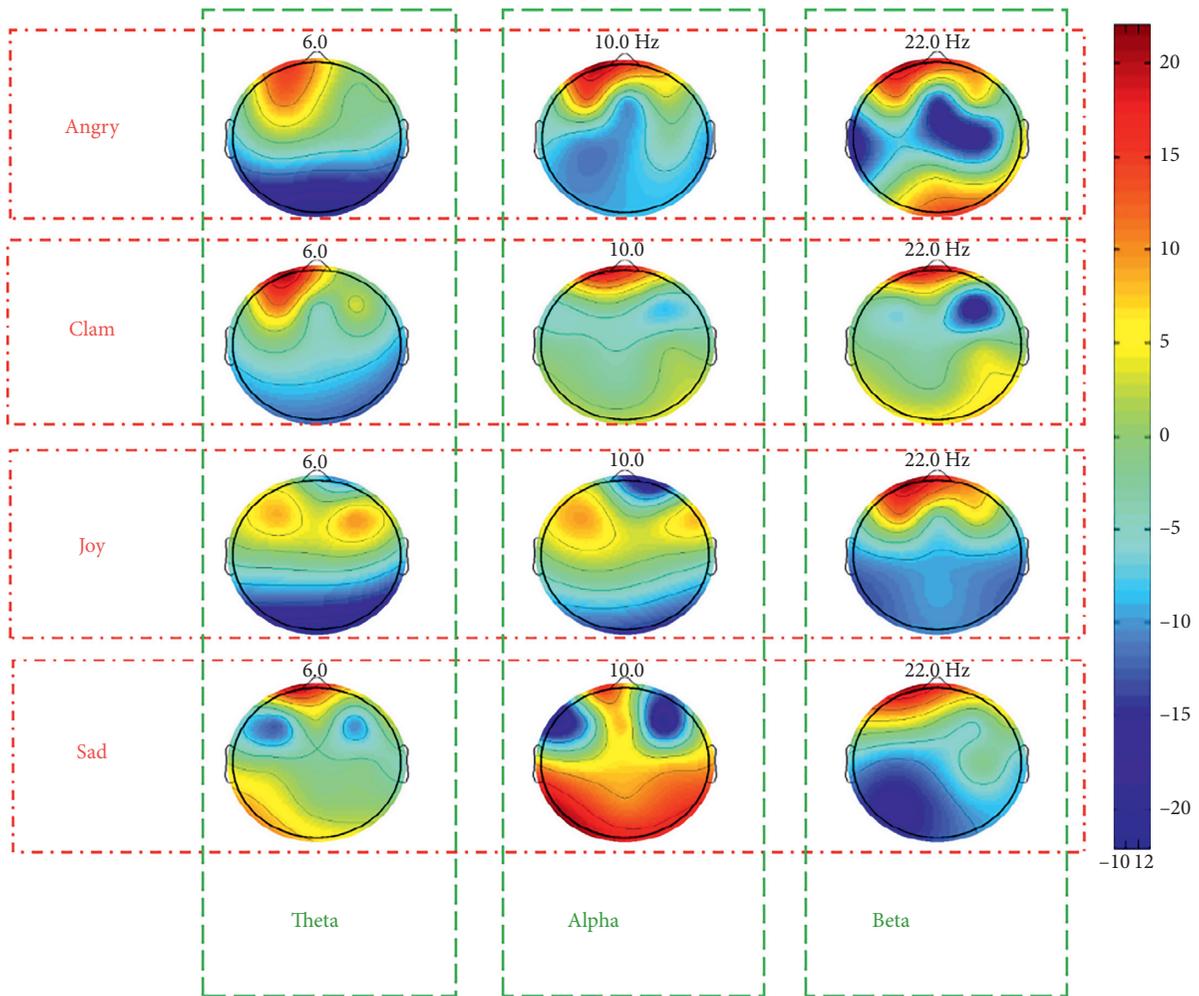


FIGURE 9: Three distinct frequency-band topographic maps distinguishing the emotions calmness, happiness, sadness, and anger evoked by music.

that the alpha band is more active when listening to emotionally evocative music.

4.2. Nonlinear Features. To analyse the nonlinear characteristics of the selected feature set, as shown in Figure 10, we constructed a frequency histogram of the selected features and studied the relationships between different emotions and nonlinear characteristics.

For the Pz electrode, from the distribution histogram of the 6 features in Figure 10, we can find some relationships between it and the emotion classifications. The value of “angry” was mostly distributed in the lower numerical segment in the histogram of maximum Lyapunov exponent. It is the same as the histogram of spectral entropy. But for “calm”, numerical distribution differences were not so obvious except in complexity of C0 histogram. The value of “joy” was mostly distributed in the higher numerical segment in the histogram of complexity of C0, entropy of K, and spectral entropy compared with “angry”. The value of “sad” was mostly distributed in the higher numerical segment in the histogram of approximate entropy and spectral entropy.

For the T3 electrode, as shown in Figure 11, we can find some differences between the histograms of “angry” and “sad” according to approximate entropy, entropy of K, and spectral entropy. But, the relationships between features and emotions were less obvious than for electrodes Pz and T4.

For the T4 electrode, from the distribution histogram of the 6 features in Figure 12, we can also find some relationships between it and the emotion classifications. The value of “angry” was mostly distributed in the lower numerical segment in the histogram of complexity of C0, maximum Lyapunov exponent, complexity of LZ, and spectral entropy. But for “calm”, numerical distribution differences were also not so obvious except in complexity of LZ histogram. The value of “joy” was mostly distributed in the higher numerical segment in the histogram of complexity of C0, maximum Lyapunov exponent, complexity of LZ, and spectral entropy. The value distribution of “sad” was very similar to the distribution of “joy”, except some small differences in approximate entropy and entropy of K.

By the comparison of nonlinear features’ value distribution histogram, some differences could be found between the four emotion states. Some differences were obvious such

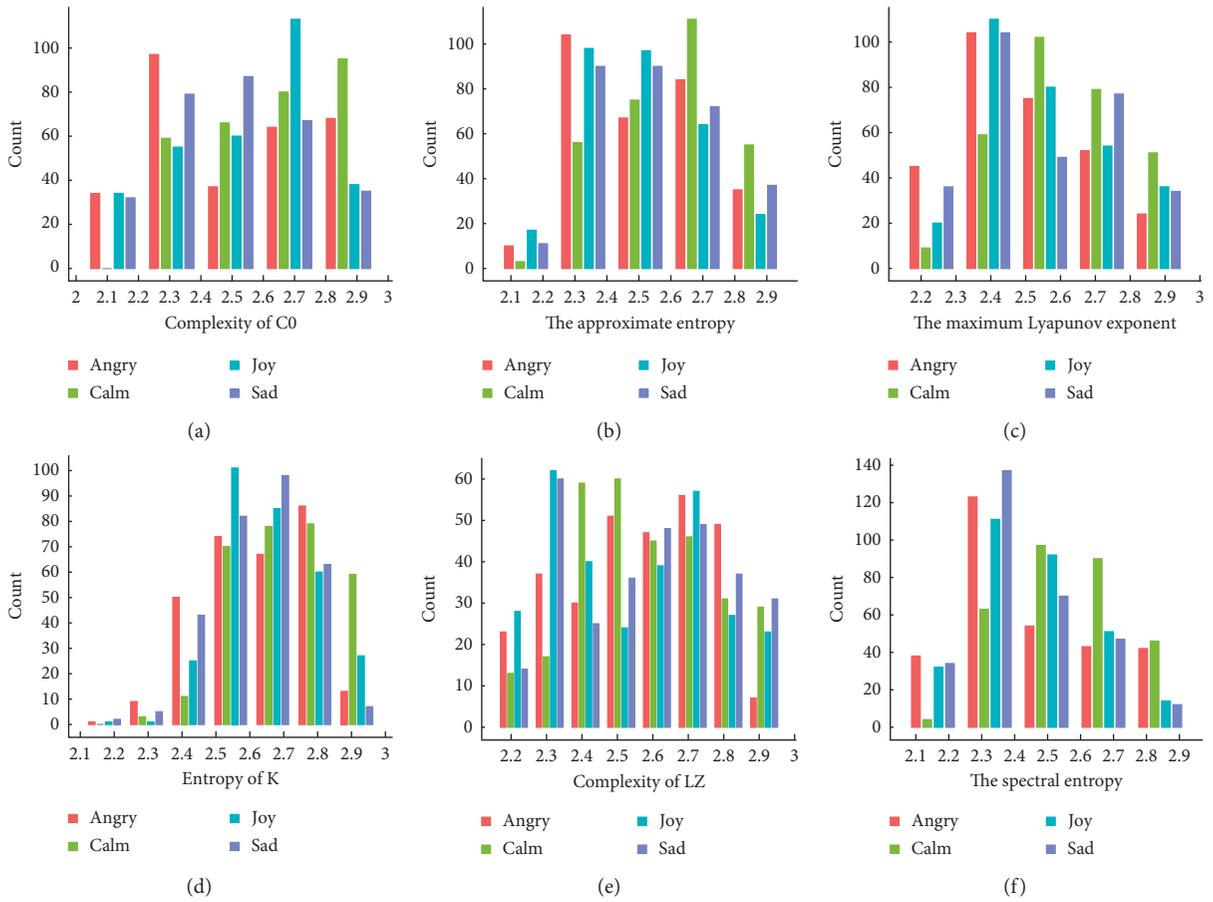


FIGURE 10: Nonlinear characteristic frequency distribution of the Pz electrode.

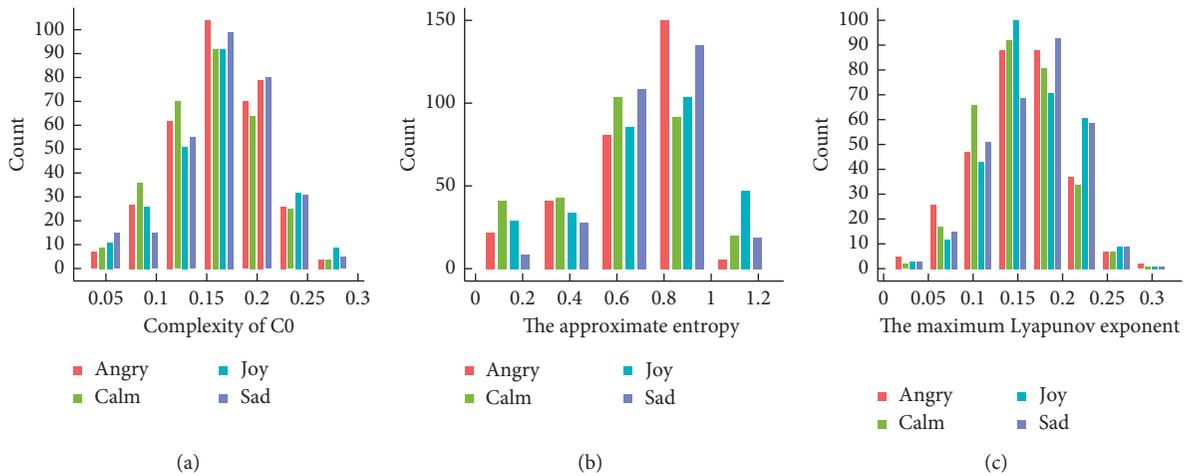


FIGURE 11: Continued.

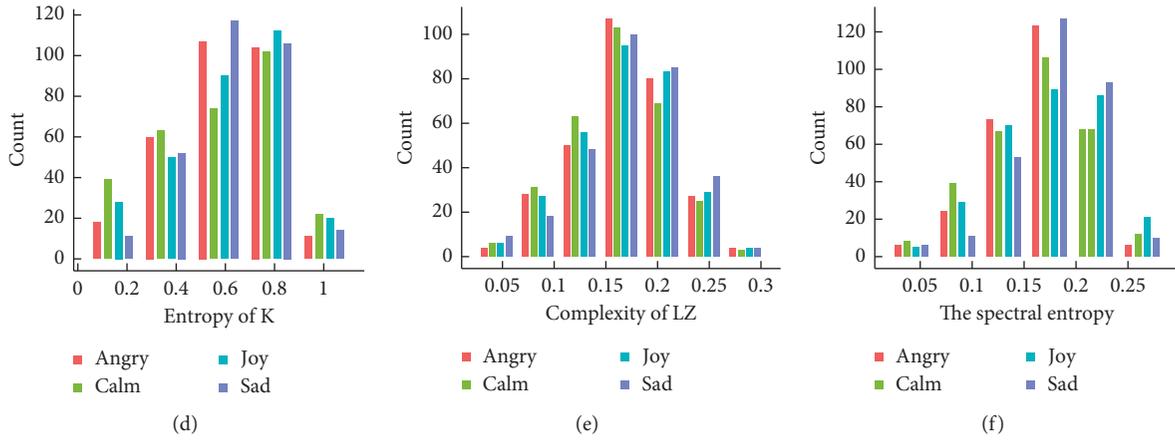


FIGURE 11: Nonlinear characteristic frequency distribution of the T3 electrode.

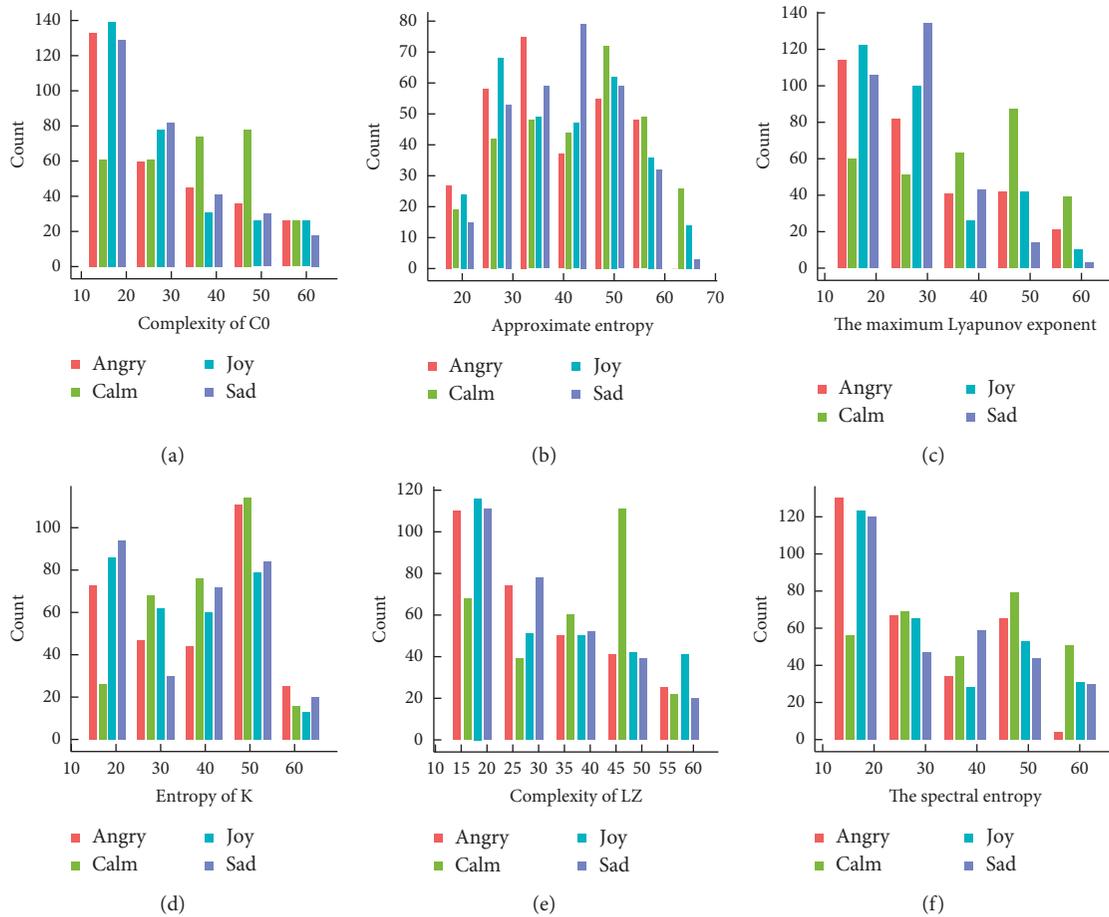


FIGURE 12: Nonlinear characteristic frequency distribution of the T4 electrode.

as “angry,” “joy,” and “sad”. But for “calm,” the differences were not so obvious.

4.3. Examination and Repeatability. We then compared emotion recognition accuracy among the 6-dimensional linear features, 6-dimensional nonlinear features, the

selected 12-dimensional features, and the 27-dimensional features using different algorithms. The recognition rate was higher for nonlinear features than linear features for all algorithms, possibly because differences in nonlinear EEG features are larger than the deviations of frequency features within the same frequency band, such as the mean values of centre frequency and maximum frequency.

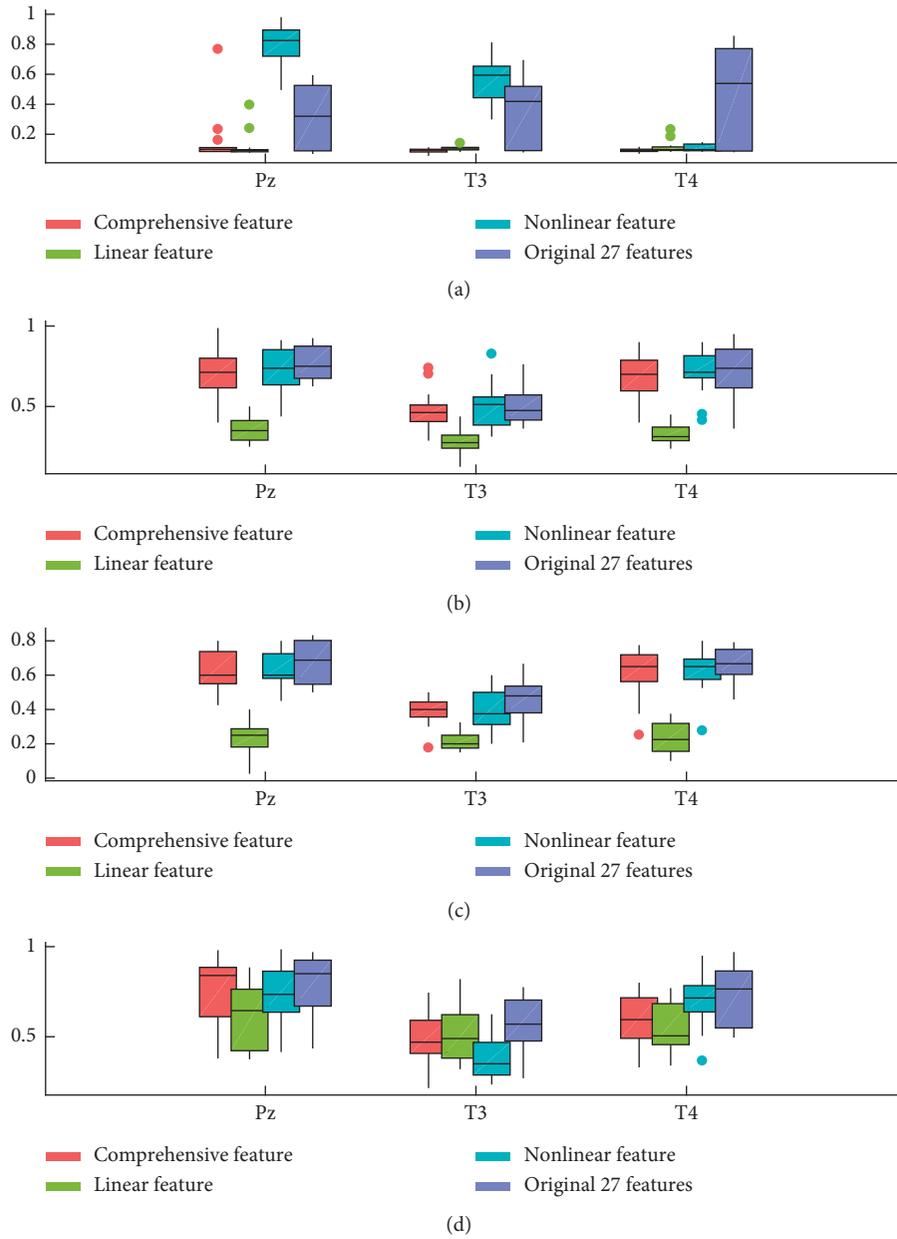


FIGURE 13: Recognition accuracy comparison among feature subsets. (a) Feature reduction recognition rate verified by SVM. (b) Feature reduction recognition rate verified by C4.5. (c) Feature reduction recognition rate verified by BP. (d) Feature reduction recognition rate verified by LDA.

Therefore, nonlinear features may be more suitable for classification of mood evoked by music. The selected features were then compared with the original 27-dimensional features. Accuracy was not significantly higher than that with the original set, possibly because these features were selected based on a comparison of all datasets across subjects and were not the best set for any individual. However, the smaller number of dimensions with equivalent accuracy indicates that redundant features were removed. Therefore, the 12-dimensional features selected can be regarded as the main EEG features distinguishing the emotions calm, anger, joy, and sadness evoked by

music. So, we selected the nonlinear features of Pz electrodes as the best feature set for emotion recognition. The results are shown in Figure 13.

To test the repeatability of the classification effectiveness of the selected feature set, we used the EEG signal of the 15 volunteers as the repeatability examination data source. The classification procedures were repeated for 10 times for each volunteer. For every one, the EEG data was selected in four emotion statures including joy, sad, calm, and angry. That was, for every emotion of every volunteer, 8 randomly selected EEG data were employed as testing data, and the other 72 EEG data were used as the training data. The classification results are

TABLE 8: Repeatability of the classification effectiveness.

1	2	3	4	5	6	7	8	9	10	Ave
100	87.5	87.5	87.5	87.5	87.5	100	87.5	100	75	90
100	100	100	100	87.5	100	87.5	87.5	87.5	75	92.5
100	87.5	87.5	100	87.5	100	100	100	100	87.5	95
87.5	87.5	75	87.5	87.5	87.5	87.5	87.5	87.5	100	87.5
87.5	87.5	87.5	100	100	100	100	87.5	87.5	100	93.75
100	87.5	87.5	100	87.5	75	87.5	100	87.5	100	91.25
100	100	100	87.5	100	100	100	100	87.5	100	97.5
87.5	87.5	75	87.5	87.5	87.5	87.5	87.5	87.5	87.5	86.25
87.5	100	75	87.5	87.5	100	87.5	87.5	75	87.5	87.5
100	100	100	100	87.5	100	100	100	100	100	98.75
87.5	100	100	87.5	100	87.5	100	87.5	100	87.5	93.75
87.5	87.5	75	87.5	87.5	87.5	87.5	87.5	75	87.5	85
87.5	87.5	87.5	75	87.5	87.5	75	87.5	87.5	87.5	85
100	87.5	100	87.5	87.5	87.5	100	87.5	87.5	87.5	91.25
87.5	87.5	87.5	100	87.5	100	87.5	87.5	87.5	100	91.25

TABLE 9: Recognition accuracy rate of the DEAP database.

Method	Valence (%)	Arousal (%)
SVM	73.45	80.20
BP	74.67	77.92
C4.5	84.91	89.65
LDA	72.33	76.68

listed in Table 8. It showed that the average recognition rates were all higher than 85% and the nonlinear features of Pz electrode were effective for emotion recognition.

The DEAP database has also been tested in this paper. We used the same data label as the article [1]. The score of valence and arousal dimension 1–3 is regarded as low group and 7–9 is high group, and finally, more than 20 EEG data evoked by videos were selected for every volunteer. The 12-dimension features mentioned in this paper were employed as the feature set. The result is that classification accuracy is higher than the statistical characteristics mentioned in the article. And as shown in Table 9, for C4.5 classifier, the correct classification rate is 84.91% and 89.65% for valence and arousal.

5. Conclusions

This study analyzed the linear and nonlinear characteristics of EEG signals recorded during music evoking distinct emotional responses (calm, joy, sadness, and anger) and identified those features most effective for accurate EEG-based recognition of emotion. The EEG characteristics of 12 EEG electrodes yielded 27 dimensions each. Statistical analysis revealed that electrodes T3, T4, and Pz were most likely to be associated with the music stimulus. The CFS method was used to identify those features most effective for EEG-based emotion recognition without redundancy. We then used different classifiers to test whether the selected feature set was more accurate than the original feature data. The results can be summarised as follows:

- (1) The algorithms C4.5 are more effective for emotion classification of EEG signals than LDA, BP neural network, and SVM.

- (2) We used brain topographic maps and frequency distribution histograms to identify the optimal subset of linear and nonlinear features of the three electrodes and identified six-dimensional linear features (centre frequency of the alpha band, mean theta band, centre frequency of the theta wave, mean delta band, centre frequency of the delta band, and maximum power of the delta wave) plus six-dimensional nonlinear features (entropy of K, approximate entropy, maximum Lyapunov exponent, C0 complexity, spectral entropy, and LZ complexity). These dimensions may be the most representative features for the classification of music-evoked emotions.
- (3) We compared discrimination accuracy among the different combinations of linear and nonlinear features and found that the nonlinear features were more effective. Finally, the 12 selected dimensional features were as accurate as the original 27 dimensions, indicating that redundant features were eliminated. Then the classification results of 10 times randomly examinations, it could be concluded that the selected 6 nonlinear features of Pz are more effective than other features when used as the key feature set to classify human emotion statuses.

Data Availability

The music stimuli used in this study are from a database of 1000 songs, which was selected by Mohammad Soleymani and colleagues of the University of Geneva for emotion analysis from the Free Music Archive (<http://freemusicarchive.org/>).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] K. Xie, *The EEG Recognition Algorithms of to Music Interferenceemotional Music*, University of Electronic Science and Technology of China, Chengdu, China, 2013.
- [2] J. Jeong, "EEG dynamics in patients with Alzheimer's disease," *Clinical Neurophysiology*, vol. 115, no. 7, pp. 1490–1505, 2004.
- [3] N. Thammasan, K. Moriyama, K.-I. Fukui, and M. Numao, "Familiarity effects in EEG-based emotion recognition," *Brain Informatics*, vol. 4, no. 1, pp. 39–50, 2016.
- [4] Y. Kumagai, M. Arvaneh, and T. Tanaka, "Familiarity affects entrainment of EEG in music listening," *Frontiers in Human Neuroscience*, vol. 11, p. 384, 2017.
- [5] H. Santosa, M. J. Hong, and K.-S. Hong, "Lateralization of music processing with noises in the auditory cortex: an fNIRS study," *Frontiers in Behavioral Neuroscience*, vol. 8, p. 418, 2014.
- [6] K.-S. Hong and H. Santosa, "Decoding four different sound-categories in the auditory cortex using functional near-infrared spectroscopy," *Hearing Research*, vol. 333, pp. 157–166, 2016.
- [7] M. Bigliassi, C. I. Karageorghis, A. V. Nowicky, M. J. Wright, and G. Orgs, "Effects of auditory distraction on voluntary movements: exploring the underlying mechanisms associated

- with parallel processing,” *Psychological Research*, vol. 82, no. 4, pp. 720–733, 2017.
- [8] J. A. Lopata, E. A. Nowicki, and M. F. Joannis, “Creativity as a distinct trainable mental state: an EEG study of musical improvisation,” *Neuropsychologia*, vol. 99, pp. 246–258, 2017.
- [9] S. W. Phneah and H. Nisar, “EEG-based alpha neurofeedback training for mood enhancement,” *Australasian Physical & Engineering Sciences in Medicine*, vol. 40, no. 2, pp. 325–336, 2017.
- [10] D. A. Bridwell, E. Leslie, D. Q. McCoy, S. M. Plis, and V. D. Calhoun, “Cortical sensitivity to guitar note patterns: EEG entrainment to repetition and key,” *Frontiers in Human Neuroscience*, vol. 11, 2017.
- [11] A. Martínez-Rodrigo, A. Fernández-Sotos, J. M. Latorre, J. Moncho-Bogani, and A. Fernández-Caballero, “Neural correlates of phrase rhythm: an EEG study of bipartite vs. Rondo sonata form,” *Frontiers in Neuroinformatics*, vol. 11, 2017.
- [12] L. Zhou, F. Liu, X. Jing, and C. Jiang, “Neural differences between the processing of musical meaning conveyed by direction of pitch change and natural music in congenital amusia,” *Neuropsychologia*, vol. 96, pp. 29–38, 2017.
- [13] J. Lu, D. Wu, H. Yang, C. Luo, C. Li, and D. Yao, “Scale-free brain-wave music from simultaneously EEG and fMRI recordings,” *PLoS One*, vol. 7, no. 11, Article ID e49773, 2012.
- [14] G.-Q. Di, M.-C. Fan, and Q.-H. Lin, “An experimental study on EEG characteristics induced by intermittent pure tone stimuli at different frequencies,” *Applied Acoustics*, vol. 141, no. 6, pp. 46–53, 2018.
- [15] V. Kurbalija, M. Ivanović, M. Radovanović, Z. Geler, W. Dai, and W. Zhao, “Emotion perception and recognition: an exploration of cultural differences and similarities,” *Cognitive Systems Research*, vol. 52, no. 6, pp. 103–116, 2018.
- [16] N. Kobayashi and M. Nakagawa, “BCI-based control of electric wheelchair using fractal characteristics of EEG,” *IEEE Transactions on Electrical and Electronic Engineering*, vol. 13, no. 12, pp. 1795–1803, 2018.
- [17] G. Zhao, Y. Zhang, and Y. Ge, “Frontal EEG asymmetry and middle line power difference in discrete emotions,” *Frontiers in Behavioral Neuroscience*, vol. 12, no. 11, pp. 1–14, 2018.
- [18] M. Ivanovic, Z. Budimac, M. Radovanovic et al., “Emotional agents-state of the art and applications,” *Computer Science and Information Systems*, vol. 12, no. 4, pp. 1121–1148, 2015.
- [19] H. J. Yoon and S. Y. Chung, “EEG-based emotion estimation using Bayesian weighted-log-posterior function and perceptron convergence algorithm,” *Computers in Biology and Medicine*, vol. 43, no. 12, pp. 2230–2237, 2013.
- [20] X. Li, B. Hu, S. Sun, and H. Cai, “EEG-based mild depressive detection using feature selection methods and classifiers,” *Computer Methods and Programs in Biomedicine*, vol. 136, pp. 151–161, 2016.
- [21] J. L. Xu and G. K. Zuo, “Motion algorithm for EEG feature selection based on mutual information and principal component analysis,” *Journal of Biomedical Engineering*, vol. 33, no. 2, pp. 201–207, 2016.
- [22] H. Li, *EEG Signal Analysis Technology and Neural Mechanism Research about Music Emotion*, Harbin Institute of Technology, Harbin, China, 2018.
- [23] L. Zhong, *Emotion Recognition Based on Multiple Physiological Signals*, Henan University, Kaifeng, China, 2018.
- [24] D. Shon, K. Im, J.-H. Park, D.-S. Lim, B. Jang, and J.-M. Kim, “Emotional stress state detection using genetic algorithm-based feature selection on EEG signals,” *International Journal of Environmental Research and Public Health*, vol. 15, no. 11, pp. 2461–11, 2018.
- [25] M. Soleymani, M. N. Caro, E. M. Schmidt et al., “1000 songs for emotional analysis of music,” in *Proceedings of ACM International Workshop on Crowdsourcing for Multimedia*, pp. 1–6, ACM, Orlando, FL, USA, November 2013.
- [26] J. A. Russell, “Measures of emotion,” in *The Measurement of Emotions*, Academic Press, Cambridge, MA, USA, 1989.
- [27] W. J. Han, H. F. Li, H. B. Ruan et al., “A review of research on speech and emotional recognition,” *Journal of Software*, vol. 25, no. 1, pp. 37–50, 2014.
- [28] L. Duan, H. Zhong, J. Miao, Z. Yang, W. Ma, and X. Zhang, “A voting optimized strategy based on ELM for improving classification of motor imagery BCI data,” *Cognitive Computation*, vol. 6, no. 3, pp. 477–483, 2014.
- [29] Q. Cai, P. H. Chen, and W. Wei, “EEG signal analysis using wavelet transform and its application,” *Journal of Huaqiao University*, vol. 36, no. 2, pp. 166–170, 2015.
- [30] Y. J. Lu, L. L. Dai, H. Z. Wu et al., “EEG study on the relaxation of sadness in different types of music,” *Psychological exploration*, vol. 32, no. 4, pp. 369–375, 2012.
- [31] J. Yang, J. Qin, H. Cai et al., “Outcome anticipation and appraisal during risk decision making in heroin addicts,” in *Proceedings of Revised Selected Papers of the Second International Conference on Human Centered Computing*, pp. 534–543, Springer-Verlag New York, Inc., Colombo, Sri Lanka, January 2016.
- [32] Q. Zhao, B. Hu, L. Liu et al., “An EEG based nonlinearity analysis method for schizophrenia diagnosis,” in *Proceedings of International Conference on Biomedical Engineering*, Penang, Malaysia, February 2012.
- [33] D. Li, M. Ni, and S. Dun, “Phase-amplitude coupling in human scalp EEG during NREM sleep,” in *Proceedings of International Conference on Biomedical Engineering and Informatics*, pp. 219–223, IEEE, Datong, China, October 2016.
- [34] S. M. Pincus, “Approximate entropy as a measure of system complexity,” *Proceedings of the National Academy of Sciences*, vol. 88, no. 6, pp. 2297–2301, 1991.
- [35] J. S. Richman and J. R. Moorman, “Physiological time-series analysis using approximate entropy and sample entropy,” *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 278, no. 6, pp. H2039–H2049, 2000.
- [36] P. Grassberger and I. Procaccia, “Measuring the strangeness of strange attractors,” *Physica D: Nonlinear Phenomena*, vol. 9, no. 1–2, pp. 189–208, 1983.
- [37] C. Liu, J. Jiang, F. Liu, and X. Qiu, “Quantitative determination on transfixion of joints by Kolmogorov entropy theory and its application to sandstone,” *Chinese Journal of Geotechnical Engineering*, vol. 29, no. 11, pp. 1730–1732, 2007.
- [38] L. Liu and T. Wang, “Comparison of TOPS strings based on LZ complexity,” *Journal of Theoretical Biology*, vol. 251, no. 1, pp. 159–166, 2008.
- [39] A. Dabrowski, “The largest transversal Lyapunov exponent and master stability function from the perturbation vector and its derivative dot product (TLEVP),” *Nonlinear Dynamics*, vol. 69, no. 3, pp. 1225–1235, 2012.
- [40] V. K. Goel, J. M. Winterbottom, K. Schulte et al., “Ligamentous laxity across C0-C1-C2 complex,” *Spine*, vol. 15, no. 10, pp. 990–996, 1990.
- [41] E. Bozzo, R. Carniel, and D. Fasino, “Relationship between singular spectrum analysis and Fourier analysis: theory and application to the monitoring of volcanic activity,” *Computers & Mathematics with Applications*, vol. 60, no. 3, pp. 812–820, 2010.

- [42] Y. Dai, H. Zhang, X. Mao, and P. Shang, "Complexity-entropy causality plane based on power spectral entropy for complex time series," *Physica A: Statistical Mechanics and its Applications*, vol. 509, no. 11, pp. 501–514, 2018.
- [43] M. A. Hall, *Correlation-based feature selection for machine learning*, Ph.D. thesis, The University of Waikato, Hamilton, New Zealand, 1999.
- [44] V. Bolón-Canedo, I. Porto-Díaz, N. Sánchez-Marroño, and A. Alonso-Betanzos, "A framework for cost-based feature selection," *Pattern Recognition*, vol. 47, no. 7, pp. 2481–2489, 2014.
- [45] X. Li, B. Hu, J. Shen, T. Xu, and M. Retcliffe, "Mild depression detection of college students: an EEG-based solution with free viewing tasks," *Journal of Medical Systems*, vol. 39, no. 12, pp. 1–6, 2015.
- [46] S. N. Daimi and G. Saha, "Classification of emotions induced by music videos and correlation with participants' rating," *Expert Systems with Applications*, vol. 41, no. 13, pp. 6057–6065, 2014.
- [47] M. B. Küssner, A. M. B. D. Groot, W. F. Hofman et al., "EEG beta power but not background music predicts the recall scores in a foreign-vocabulary learning task," *PLoS One*, vol. 11, no. 8, Article ID e0161387, 2016.

Research Article

Motor Imagery EEG Classification Based on Decision Tree Framework and Riemannian Geometry

Shan Guan , Kai Zhao , and Shuning Yang 

School of Mechanical Engineering, Northeast Electric Power University, 132012 Jilin, China

Correspondence should be addressed to Shan Guan; guanshan1970@163.com

Received 24 September 2018; Revised 20 December 2018; Accepted 3 January 2019; Published 21 January 2019

Guest Editor: Anastassia Angelopoulou

Copyright © 2019 Shan Guan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a novel classification framework and a novel data reduction method to distinguish multiclass motor imagery (MI) electroencephalography (EEG) for brain computer interface (BCI) based on the manifold of covariance matrices in a Riemannian perspective. For method 1, a subject-specific decision tree (SSDT) framework with filter geodesic minimum distance to Riemannian mean (FGMDRM) is designed to identify MI tasks and reduce the classification error in the nonseparable region of FGMDRM. Method 2 includes a feature extraction algorithm and a classification algorithm. The feature extraction algorithm combines semisupervised joint mutual information (*semi*-JMI) with general discriminate analysis (GDA), namely, SJGDA, to reduce the dimension of vectors in the Riemannian tangent plane. And the classification algorithm replaces the FGMDRM in method 1 with k-nearest neighbor (KNN), named SSDT-KNN. By applying method 2 on BCI competition IV dataset 2a, the kappa value has been improved from 0.57 to 0.607 compared to the winner of dataset 2a. And method 2 also obtains high recognition rate on the other two datasets.

1. Introduction

Brain computer interface (BCI) based on motor imagery (MI) is used to analyze human intention by electroencephalogram (EEG) signals generated by human brain electrophysiological activity [1, 2]. Based on BCI technology, exoskeletons can be used to help people with physical disabilities regain their motor ability, and BCI also has wide applications in smart home, entertainment, military, and other fields [3–6].

Common spatial pattern (CSP) is widely used in motor imagery to extract EEG features [7]. CSP has excellent performance in two classification tasks, but the drawback is that it needs a lot of electrodes [8].

Despite its short history, the use of the Riemannian geometry in BCI decoding is currently attracting increasing attention [9–13]. Covariance matrices lie in the space of symmetric positive definite (SPD) matrices, which can be formulated as a Riemannian manifold [14]. In the BCI field, the connections of the CSP algorithm and the tools of information geometry have been investigated, considering

several divergence functions in alternative to the Riemannian distance [15–18]. Barachant et al. proposed a simple data augmentation approach for improving the performance of the Riemannian mean distance to mean (MDM) algorithm [13]. Kumar et al. propose a single band CSP framework for MI-BCI that utilizes the concept of tangent space mapping in the manifold of covariance matrices, and the proposed method obtains good results when compared to other competing methods [19]. A hierarchical MDM classifier for multiclass problem has been tested in [20].

Advanced classifiers based on the tangent space on the Riemannian manifold of positive matrices are also receiving increasing attention. Barachant et al. map the covariance matrices in the tangent space and apply feature selection and linear discriminate analysis (LDA) in the tangent space [10]. For the application of the classifier in the tangent space, the problem is that the curse of dimensionality. Traditional data dimensionality reduction methods include two categories: linear dimensionality reduction (LDR) and nonlinear dimensionality reduction (NLDR). Since most of the actual

data are nonlinear, NLDR techniques such as locally linear embedding (LLE) [21], isometric mapping (ISOMAP) [22], maximum variance unfolding (MVU) [23], and t-distributed stochastic neighbor embedding (t-SNE) [24, 25] are used to tackle problems widely. Lee et al. used discrete wavelet transform (DWT) and continuous wavelet transform (CWT) to extract features of MI tasks, and Gaussian mixture model (GMM) was used to construct GMM supervectors; this method accelerates the speed of training and improves the accuracy of motor imagery [26]. Sadatnejad et al. propose a new kernel to preserve the topology of data points in the feature space, and the proposed kernel is strong, particularly in the cases where data points have a complex and nonlinear separable distribution [8]. Xie et al. proposed a framework for intrinsic submanifold learning from a high-dimensional Riemannian manifold; the proposed method exhibited strong robustness against a small training dataset [27].

There is still another approach for overcoming the problem of high dimensionality in SPD manifolds. And this method maps from a high-dimensional SPD manifold to a lower dimensional one while the geometry of SPD manifolds is preserved. And there are only two works of this way. Davoudi et al. [14] proposed distance preservation to local mean (DPLM) as dimensionality reduction technique, combined with FGMDM, the best performance of this article in terms of kappa value is 0.60. Harandi et al. [28] learned a mapping that maximizes the geodesic distances between interclass and simultaneously minimizes the distances between intraclass, and it is done via an optimization on Grassmann manifolds.

In this paper, we proposed a novel SSDT-FGMDRM and SSDT-KNN for the classification of multiclass MI tasks by designing a simple yet efficient subject-specific decision tree framework. Method 1 contains SSDT-FGMDRM to improve the performance of FGMDRM. For each individual, method 1 first separates the two most discriminative classes from the group. Furthermore, the remaining categories including the misclassification samples of the previous nodes are reclassified in the last node. Method 2 contains SSDT-KNN and a NLDR method named SJGDA. SJGDA combines the advantage of *semi*-JMI and GDA, and method 2 performed well on different datasets. The aims of this article are as follows:

- (1) To verify the effectiveness of the proposed SSDT framework through dataset 1
- (2) To verify the superiority of SJGDA in feature extraction, compared with *semi*-JMI and GDA
- (3) To validate the generalization ability of method 2 through different datasets, in this paper

The rest of the paper is organized as follows: Section 2 introduced the mathematical preliminaries of the Riemannian geometry. Section 3 discussed the proposed methods in detail. Three datasets are introduced in Section 4. The results of our work are discussed in Section 5. And in Section 6, we compared our methods with the state of the art. This paper concludes in Section 7.

2. Geometry of SPD Matrices

Let X_i represent a short segment of continuous EEG signals, and X_i can be denoted as follows:

$$X_i = [X_{t+T_i} \cdots X_{t+T_i+T_s-1}] \in \mathbb{R}^{n \times T_s}, \quad (1)$$

where X_i corresponds to the i th trail of imaged movement starting at time $t = T_i$. T_s denotes the number of sampled points of the selected segment.

For the i th trail, the spatial covariance matrix (SCM) $P_i \in \mathbb{R}^{n \times n}$ can be calculated as follows:

$$P_i = \frac{1}{T_s - 1} X_i X_i^T. \quad (2)$$

Based on the SCM, there are two ways to classify MI tasks in the Riemannian manifold.

2.1. Filter Geodesic Minimum Distance to the Riemannian Mean. The Riemannian distance between two SPD matrices P_1 and P_2 in $P(n)$ is given by [29]

$$\delta_R(P_1, P_2) = \left\| \log(P_1^{-1} P_2) \right\|_F = \left[\sum_{i=1}^n \log^2 \lambda_i \right]^{1/2}. \quad (3)$$

Given m SPD matrices P_1, \dots, P_m , the geometric mean in the Riemannian sense is defined as

$$\mathfrak{M}(P_1, \dots, P_m) = \arg \min_{P \in P(n)} \sum_{i=1}^m \delta_R^2(P, P_i). \quad (4)$$

For algorithm mean Riemannian distance to Riemannian mean (MDRM), we compute the Riemannian distance between unknown class P to the Riemannian mean point of each class and classify the unknown class into categories corresponding to the shortest distance. Inspired by the principal geodesics analysis (PGA) method [30], the literature [31] finds a set of filters by applying an extension of Fisher linear discriminant analysis (FLDA) named Fisher geodesic discriminant analysis (FGDA). And then, apply these filters to MDRM to form filter geodesic minimum distance to Riemannian mean (FGMDRM). More details can be seen from [31].

2.2. Tangent Space Mapping. As shown in Figure 1, the SPD matrix of P is denoted by a differentiable Riemannian manifold Z . Each tangent vector S_i can be seen as the derivative at $t=0$ of the geodesic $\Gamma(t)$ between P and the exponential mapping $P_i = \text{EXP}_P(S_i)$, defined as follows:

$$\text{Exp}_P(S_i) = P_i = P^{1/2} \exp(P^{-1/2} S_i P^{-1/2}) P^{1/2}. \quad (5)$$

The inverse mapping is given by the logarithmic mapping and can be defined as follows:

$$\log_P(P_i) = S_i = P^{1/2} \log(P^{-1/2} P_i P^{-1/2}) P^{1/2}. \quad (6)$$

Using the Riemannian geodesic distance, the Riemannian mean of $I > 1$ SPD matrices by

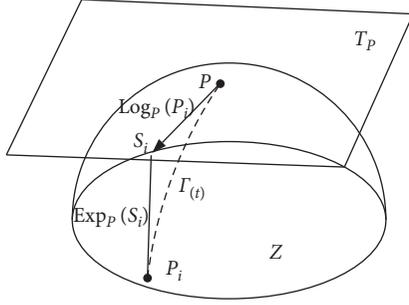


FIGURE 1: The tangent space at point P , and the geodesic $\Gamma(t)$ between P and P_i .

$$\mathfrak{P}(P_1, \dots, P_I) = \arg \min_{P \in P(n)} \sum_{i=1}^I \delta_R^2(P, P_i). \quad (7)$$

Using the tangent space located at the geometric mean of the whole set trials, $P_{\mathfrak{P}} = \mathfrak{P}(P_i, i = 1 \dots I)$, and then, each SCM P_i is mapped into this tangent space, to yield the set of $m = n(n+1)/2$ dimensional vectors:

$$S_i = \text{upper}(P_{\mathfrak{P}}^{-1/2} \log(P_i) P_{\mathfrak{P}}^{-1/2}). \quad (8)$$

Many efficient classification algorithms can be implemented in the Riemannian space [10].

3. Methods

3.1. Subject-Specific Decision Tree Framework. Decision tree is a common machine learning method. Each node of decision tree can be defined as a rule. Guo and Gelfand [32] proposed classification trees with neural network, and this method embeds multilayer neural networks directly in nodes. In the decision tree, one of the most important things is to construct a proper binary tree structure; the upper nodes have the greater impact of the accuracy of the whole samples [33]. In order to solve the multiclassification problem in this paper, we constructed a subject-specific decision tree (SSDT) classification framework as shown in Figure 2 according to the best separating principle [34]. As can be seen from Figure 2, the SSDT proposed in this paper trains a different classification model at different nodes of the decision tree.

The advantages of the SSDT framework are as follows:

- (1) This model separates the two MI tasks (e.g., C.1 and C.2) with the highest recognition rate as far as possible
- (2) At the last node, we reclassify some samples to enhance the classification ability of the classifier

3.2. Method 1: A Direct Classification Method Based on SSDT-FGMDRM. Firstly, we point out one problem of the multi-class FGMDRM by using an example. Figure 3 gives a three-class classification problem. Figure 3(a) shows the classification progress by FGMDRM. We can see that three Riemannian mean points (RMPs) are located on the manifold. Since the classification criterion is decided by the distance calculated

between the test point and the RMP, it caused a wrong classification. Figure 3(b) shows the example of the classification results obtained by using the first node of the SSDT-FGMDRM framework. It can be seen that the error classification is corrected by using the decision tree framework.

Method 1 is used to classify four types of MI tasks directly. The training and testing diagram is shown in Figure 4.

3.3. Feature Extraction Algorithm Based on the Riemannian Tangent Space. In this paragraph, we propose a novel data reduction method which combines *semi*-JMI and GDA, namely SJGDA, to solve the dimension disaster problem after tangent space mapping.

3.3.1. Semisupervised Joint Mutual Information.

Semisupervised dataset $D = D\{D_L \cup D_U\}$ consists of two parts, $D_L = \{x^i, y^i\}_{i=1}^{N_L}$ are labelled data and $D_U = \{x^{N_L+i}\}_{i=1}^{N_U}$ are unlabelled data. A binary random variable S is introduced to determine the distribution of labelled dataset and unlabelled dataset. When $s = 1$, we record the value of y , otherwise not. In this way, the labelled set D_L comes from the joint distribution $p(x, y|s=1)$, while the unlabelled set D_U comes from the distribution $p(x|s=0)$. The underlying mechanism S turns out to be very important for feature selection.

Feature selection method based on mutual information theory is a common feature selection method [35]. In these methods, we rank the features according to the score and select the features with higher scores. For example, by ranking the features according to their mutual information with the labels, we get the sort of correlation that is related to class labels. The characteristics of the score are defined as follows:

$$J_{\text{JMI}}(X_k) = \sum_{X_j \in X_\theta} \hat{I}\left(X_k; \frac{Y}{X_j}\right), \quad (9)$$

where X_θ represents the set of the features already selected and X_k is the feature ranked by scores. Y represents the label corresponding to feature X_k .

Semi-JMI is a method of using a semisupervised dataset as a training set for JMI. More details can be seen from Reference [36]. In this paper, the missingness mechanism is class-prior-change semisupervised scenario (MAR-C) [37]. After feature ranking, we can obtain a feature vector as follows:

$$f = [f_1, f_2, \dots, f_n], \quad (10)$$

where n is the length of the tangent vectors S_i . Since information redundancy exists in f , we select the best vector length m ($m < n$) of each subject by the classification recognition rate:

$$f_{\text{SJ}} = [f_1, f_2, \dots, f_m]. \quad (11)$$

3.3.2. Generalized Discriminant Analysis. After variable selection, this paper uses generalized discriminant analysis

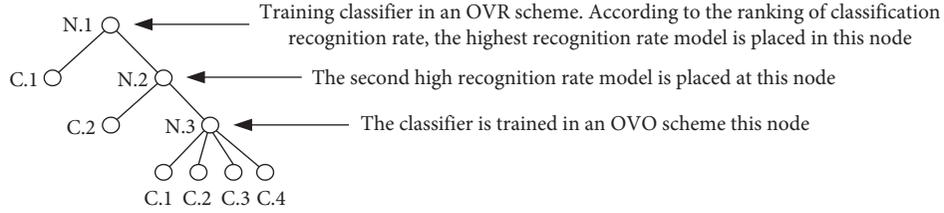


FIGURE 2: SSdT based on the best separating principle for four class. $N. i$ represents node i , and $C. i$ represents class i .

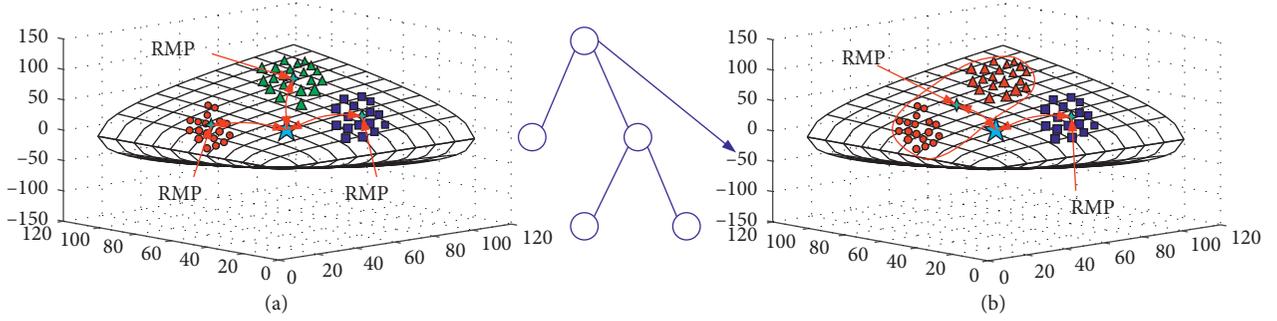


FIGURE 3: Three classification problems classified by FGMDRM (a); and a subjectspecific decision tree FGMDRM model (b).

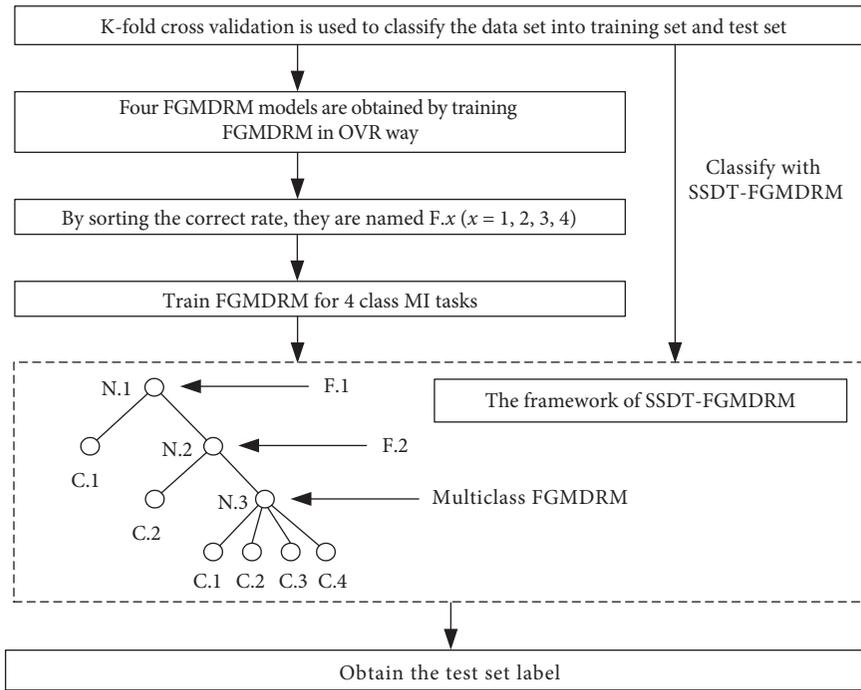


FIGURE 4: Block diagram for method 1.

(GDA) [38, 39], which is a nonlinear feature reduction technique based on kernels to reduce the length of the feature vectors f_{Sj} and their redundancies. Mapping X (f_{Sj}) into a high-dimensional space F through a kernel function Φ :

$$\begin{aligned} \Phi: R^d &\longrightarrow F, \\ x^\Phi &\longrightarrow (x). \end{aligned} \quad (12)$$

The linear Fisher decision is performed in the F space, and the criterion function for its extension is

$$J(W^\Phi) = \arg \max_{W^\Phi} \left(\frac{|(W^\Phi)^T S_B^\Phi W^\Phi|}{|(W^\Phi)^T S_T^\Phi W^\Phi|} \right), \quad (13)$$

where $W^\Phi \in F$ and S_B and S_W are between-class scatter and within-class scatter, respectively.

For the convenience of the numerical calculation, kernel functions are introduced to solve the problem:

$$k(x, y) = (\Phi(x) \cdot \Phi(y)). \quad (14)$$

Gauss kernel, poly kernel, and sigmoid kernel are widely used in GDA [40]. For test data z , its image $\Phi(z)$ in F space projects on W^Φ is as follows:

$$(W_i^\Phi \cdot \Phi(z)) = \sum_{j=1}^N \alpha_{ij} (\Phi(x_j) \cdot \Phi(z)) = \sum_{j=1}^N \alpha_{ij} k(x_j, z). \quad (15)$$

This paper uses ploy kernel to reduce the dimension. After GDA, we can get a vector f_G as follows:

$$f_G = [f_1, f_2, \dots, f_d], \quad (16)$$

where d of f_G is decided by the actual needs, and in this paper we set $d = 1$. And then, SJGDA is applied to the dataset of this paper, and the final feature vectors are constructed as follows:

$$f_{\text{SJGDA}} = [f_G, f_{\text{SJ}}]. \quad (17)$$

3.4. Method 2: SJGDA and Subject-Specific Decision Tree k -Nearest Neighbor. Method 2 is used to classify four types of MI tasks after tangent space mapping. The training and testing diagram is shown in Figure 5.

4. Description of Data

4.1. Dataset 1. BCI competition IV dataset 2a is used to evaluate the performance of the proposed two methods [41]. Dataset 2a collects 22 channel EEG data and 3 EOG channel data. Four types of motor imagery were collected: left hand, right hand, foot, and tongue. The dataset contains nine healthy subjects and each subject has two sessions, one training session and one test session. Each session has 288 trails of MI data with 72 trails for each MI task. The EEG signals are bandpass filtered by a 5-th order Butterworth filter in the 8–30 Hz frequency band. The selection of trial period is important in MI classification; we select 2 s data (0.5 s and 2.5 s) after the cue, instructing the user to perform the MI tasks by the winner of the competition.

4.2. Dataset 2. BCI competition III dataset IIIa is used to evaluate the performance of method 2. BCI III dataset IIIa contains 3 subjects: K3b, K6b, and L1b, and collects 64 channel EEG data. The EEG was sampled with 250 Hz. Four types of motor imagery were collected: left hand, right hand, foot, and tongue. More details about this dataset can be seen at Reference [42].

4.3. Dataset 3. In our own dataset, Emotiv Epoc+ is used to collect EEG data of motor imagery. It is a portable EEG acquisition device with a sampling rate of 128 Hz. It has fourteen electrode channels (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, and AF4), two inference electrodes (CMS and

DRL), and the electrode placement follows the international 10–20 standard. Equipment and the Emotiv 14 electrodes are located over 10–20 international system positions as shown in Figure 6. This experiment collected three kinds of EEG signals of one joint: imagination of shoulder flexion (F), extension (E), and abduction (A), as shown in Figure 7.

Seven subjects participated in this experimental study. These subjects were in good health. During the experiment, subjects were naturally placed with both hands, trying to avoid body or head movement. During the experiment, subjects carried out motor imagery under the outside cue, a single experiment collected EEG signal for 5 seconds, and then took 5–7 seconds to have rest, each action repeated acquisition 20 times. The experimental process is shown in Figure 8.

5. Results

5.1. Results of Method 1. We use SSDDT-FGMDRM to classify multiclass MI tasks as introduced in Section 3.1. Since there are four classes, we can have four pairs of MI tasks: left vs rest (L/RE), right vs rest (R/RE), foot vs rest (F/RE), and tongue vs rest (T/RE). For each subject, the pair with the highest accuracy is used to train N.1, and the pair with the second highest accuracy is to train N.2. Table 1 gives the ten-folder cross-validation results obtained using FGMDRM in OVR scheme.

Table 2 displays the kappa values obtained by method 1. Compared with other methods, five subjects (A03, A06, A07, A08, and A09) achieved higher kappa value of nine without exploring the frequency domain information by method 1. In the case of fixed frequency window, we have improved the mean kappa value of 0.069 than MDRM ($p = 0.4683$), and 0.139 than FGMDM_fixed ($p = 0.1423$). Our approach also shows significant improvement than FGMDM ($p = 0.6607$), which has exploited subject-specific frequency information, in terms of the kappa value of 0.039.

5.2. Results of Method 2. The results in Figure 9 show the T/RE feature distribution of the five features of subject A09. Figure 9(a) shows the first five ranked features with *semi*-JMI. After applying the *semi*-JMI, the first five best features extracted have shown statistically significant improvement in the separability with p values < 0.05 except feature 2 with p value 0.77. In Figure 9(b), the first five features extracted from primitive feature vectors with p value of 0.13, 0.05, 0.87, 0.05, and 0.13. The p values indicate that the pair T/RE have no significance in the primitive feature vectors. The results show that with our semisupervised feature ranking algorithm, the separable degree of the feature has been greatly improved.

Figure 10 shows the evolution of the classification accuracy with KNN ($k = 5$ in this paper) against the number of ranked variables in OVR scheme. L/RE and T/RE are the two pairs with the highest recognition rate, and they achieved the highest recognition rate in 100 variables. But this is still a curse of dimensionality for classifiers; GDA is used to analyze the first 100 sorted variables in our study.

As the separation of characteristics cannot meet our requirements, GDA is used to get more obvious variables.

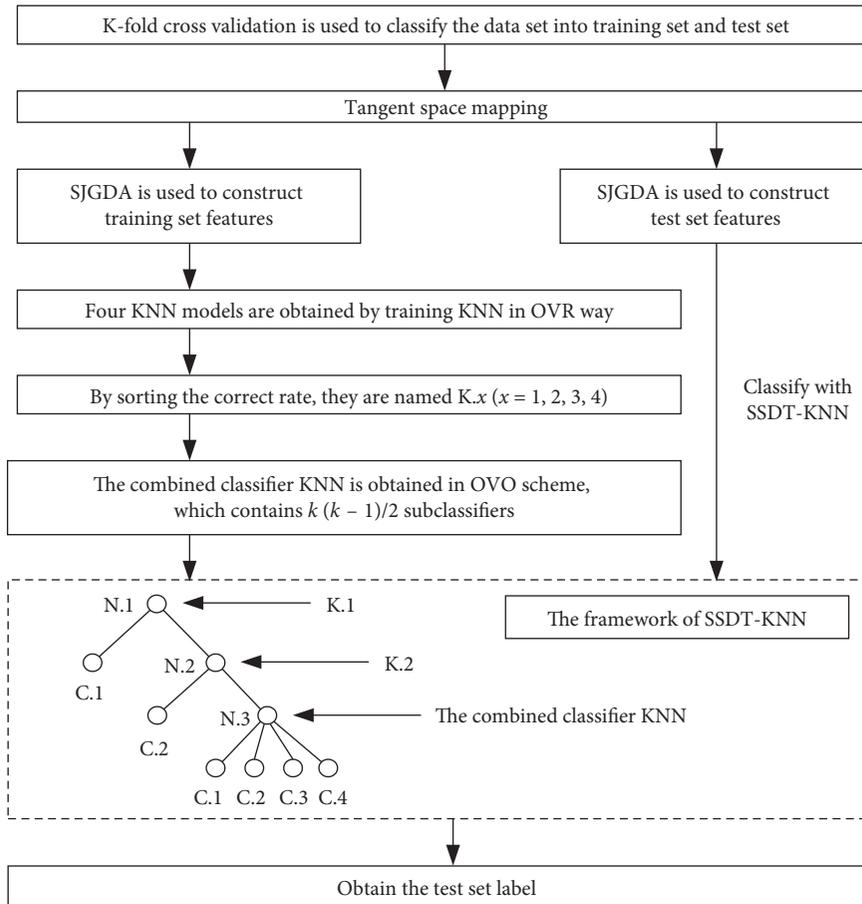


FIGURE 5: Block diagram for method 2.

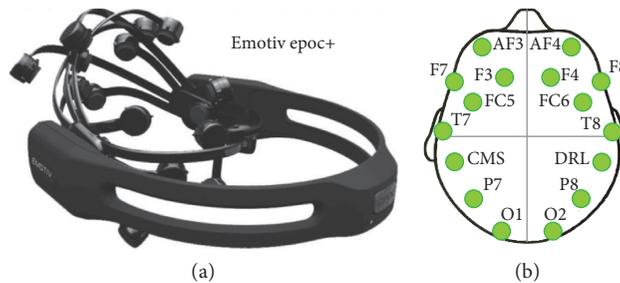


FIGURE 6: (a) Emotiv Epoc+ and (b) Emotiv 14 electrodes located over 10–20 international system positions.

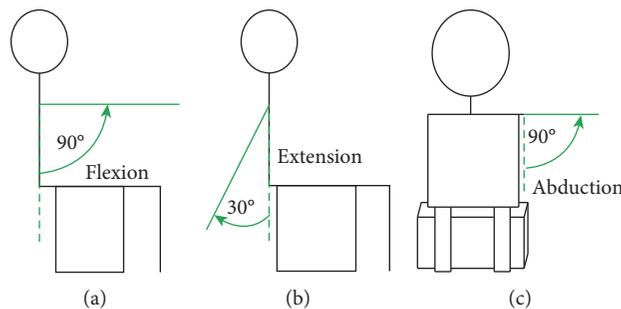


FIGURE 7: Three movements of shoulder joint: (a) flexion, (b) extension, and (c) abduction.

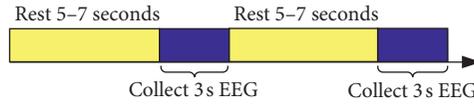


FIGURE 8: Timing for experimental process.

TABLE 1: Ten-folder cross-validation classification accuracy (%) for FGMDRM with OVR scheme applied on BCI competition dataset 2A.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	Std
L/RE	84.40	67.39	93.05	77.64	66.96	74.34	86.33	93.83	94.09	82.00	10.90
R/RE	89.26	75.33	95.16	78.56	63.92	71.81	81.18	93.39	82.92	81.28	10.24
F/RE	77.80	83.44	89.20	80.88	71.56	78.22	88.89	79.18	84.4	81.51	5.65
T/RE	88.21	68.76	90.59	79.95	71.85	76.72	90.64	94.43	94.38	83.95	9.82

TABLE 2: Kappa value comparison by SSDT-FGMDRM with other published results.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	<i>p</i> value
Method 1	0.66	0.39	0.78	0.47	0.25	0.41	0.72	0.79	0.83	0.589	
MDRM [10]	0.75	0.37	0.66	0.53	0.29	0.27	0.56	0.58	0.68	0.52	0.4683
FGMDM [14]	0.72	0.50	0.64	0.38	0.28	0.34	0.64	0.68	0.75	0.55	0.6607
FGMDM_fixed [14]	0.69	0.35	0.60	0.28	0.21	0.30	0.46	0.62	0.53	0.45	0.1423

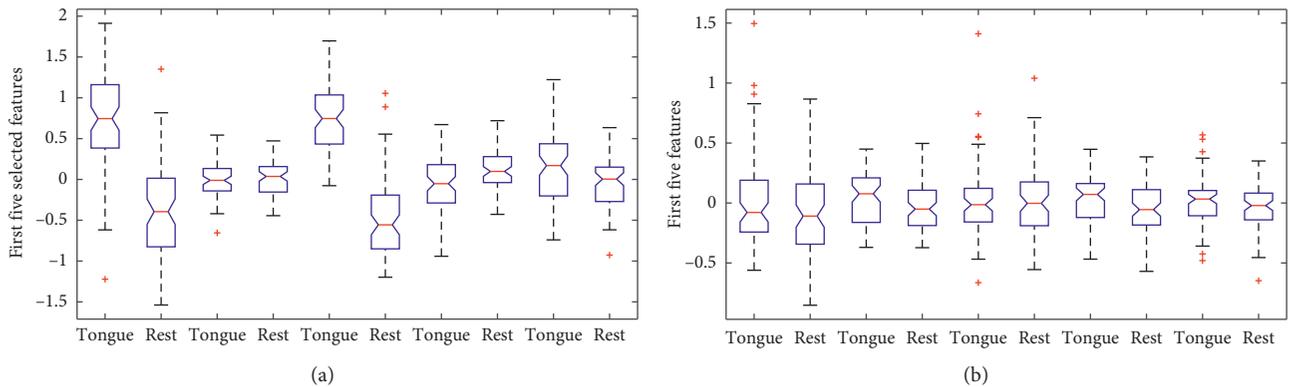


FIGURE 9: The box plot of five features of subject A09. (a) One-way ANOVA analysis on the first five features after applying *semi*-JMI. (b) One-way ANOVA analysis on the first five features from primitive vectors.

Figure 11 illustrates distributions for the first five most discriminant variables with GDA and *semi*-JMI. It can be seen from Figure 11 that L/RE is separated equally well by using GDA.

Table 3 displays ten-folder cross-validation results obtained using SJGDA and KNN in OVR scheme. It can be seen that the vectors which are mapped to the tangent space have better classification performance than that in the Riemannian manifold directly.

Table 4 presents the results obtained by SJGDA in pairwise way for multiclass MI tasks. We have six pairs of MI tasks: left and right (L/R), left and foot (L/F), left and tongue (L/T), right and foot (R/F), right and tongue (R/T), and foot and tongue (F/T).

Table 5 displays the comparison of classification accuracy using SJGDA and KNN for L/R task in 10-folder cross validation. References [8, 43–45] contain the classification of

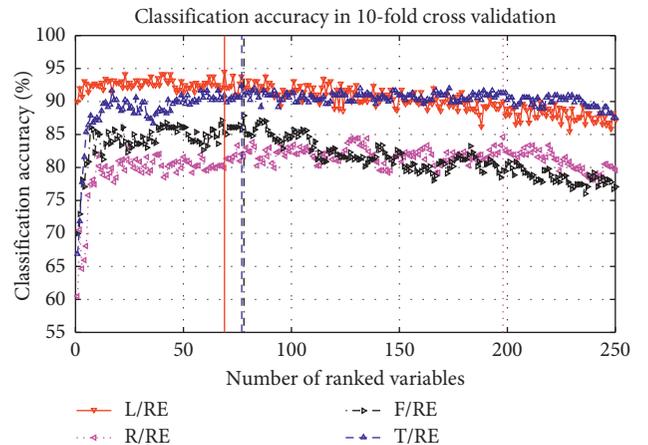


FIGURE 10: Classification accuracy corresponding the number of selected variables of Subject A09.

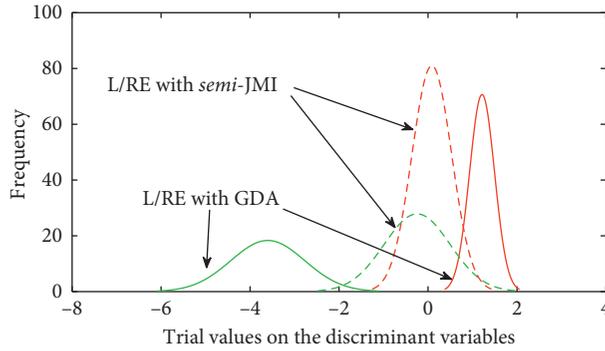


FIGURE 11: Feature distribution for the most discriminant variables in tangent space of Subject A09.

TABLE 3: Ten-folder cross-validation results (%) obtained using SJGDA and KNN in OVR scheme applied on BCI competition dataset 2A.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	Std
L/RE	90.29	74.62	93.05	75.02	75.31	78.15	85.08	92.35	95.14	84.33	8.12
R/RE	91.34	77.11	94.47	77.07	73.61	75.74	81.58	93.41	87.53	83.54	7.75
F/RE	85.06	83.70	84.77	77.11	75.36	77.09	88.17	82.30	85.44	82.11	4.24
T/RE	88.89	76.72	89.25	80.91	78.51	76.04	89.27	94.08	93.41	85.23	6.76

other publications. We have improved the accuracy compared with Reference [44] ($p=0.85$) and Reference [45] ($p=0.45$). Gaur et al. [43] ($p=0.95$) explored the specific frequency information for each subject, and Sadatnejad and Shiry Ghidary [8] ($p=0.90$) used a novel kernel for dimensionality reduction which is similar to SJGDA. Although the results in the paper are not as high as those in Reference [43], it can be concluded that there is no difference between the results in Reference [43] and those in this paper because of $p=0.95$.

Table 6 presents the results in terms of the kappa value. The proposed method 1 achieved a mean performance of 0.589 which ranks this method to the first place of the competition. And with our proposed method 2, we have achieved a mean performance of 0.607, which makes method 2 to acquire the best performance of the state of the art.

Dataset 2 is used to verify the effect of method 2, and the classification results are given directly in this paper. The results are shown in Table 7. As can be seen from Table 7, method 2 obtained the second highest recognition rate in the comparative literature. Compared with the recent reference [47], method 2 achieved good classification results.

5.3. Results of Dataset 3. Dataset 3 is used to evaluate the performance of method 2. Figure 12 shows the classification error with KNN against the number of ranked variables in OVR scheme. A/RE and F/RE are the two pairs with the lowest classification error, and they all achieved the highest recognition rate within 60 variables. In this paper, the first 60 ranked variables are used for the next analysis.

Figure 13 displays 5-folder cross-validation results obtained by using SJGDA and KNN in OVR and OVO scheme. This Figure 13(a) illustrates three possible pairs of MI tasks (F/RE, E/RE, and A/RE) for each subject. It can be learned

from the figure that flexion and abduction are the easiest movement to distinguish in six subjects of seven, and the six subjects are S1, S3, S4, S5, S6, and S7. However, due to individual differences, the highest recognition rate of each subject is different.

We also compared three possible pairs (F/E, F/A, and E/A) in OVO scheme of seven subjects. Figure 13(b) depicts the comparison results for each subject, and it can be seen that the pair of F/A obtained the highest recognition rate in seven subjects. Combined with the analysis results of Figures 13(a) and 13(b), it can be considered that flexion and extension are more obvious in the three MI tasks.

As SJGDA is a new method proposed in this paper, we also compared the feature distribution of SJGDA, GDA, and *semi*-JMI to illustrate the effectiveness of SJGDA. Figure 14 depicts the feature distribution of F/E MI tasks of seven subjects. The blue and red circles represent the two different feature classes. As shown in Figure 14, the F/E MI tasks learned by SJGDA have high separability than GDA and *semi*-JMI.

The performance of the proposed method 2 is evaluated by using classification accuracy. Since there are three classes, the chance level is 33.33%. Figure 15 demonstrates that the proposed method achieves higher performance for six subjects (S1, S2, S3, S4, S5, and S6) out of seven except S7 compared to *semi*-JMI and GDA methods. In addition, it also can be seen that GDA obtains a better classification accuracy for four subjects of seven (S1, S2, S5, and S7) compared with *semi*-JMI. The reasons for this phenomenon can be attributed to as follows: In the process of feature selection, we manually select feature dimensions suitable for classifiers, which results in partial information loss. As a feature dimensionality reduction technique, GDA is suitable for the preservation of useful information from the primitive vectors. And the proposed method SJGDA in this paper not only preserves the advantages of GDA but also adds some

TABLE 4: Ten-folder cross-validation results (%) obtained using SJGDA and KNN in OVO scheme applied on BCI competition dataset 2A.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	Std
L/R	90.95	67.29	94.52	63.87	64.48	70.33	70.29	97.95	95.00	79.41	13.85
L/F	95.19	87.43	93.93	81.84	68.10	77.14	98.57	88.29	93.81	87.14	9.29
L/T	96.52	64.24	96.57	83.24	73.67	69.91	97.95	97.23	99.29	86.51	13.16
R/F	95.90	87.52	95.05	84.84	66.71	72.88	97.24	93.18	87.43	86.75	10.01
R/T	99.33	79.82	96.48	79.12	72.22	71.48	97.24	95.05	92.29	87.00	10.61
F/T	83.94	84.17	86.62	75.10	62.62	74.21	88.31	93.10	90.33	82.04	9.10

TABLE 5: Comparison of classification accuracy (%) for L/R task with other published results using OVO scheme applied on BCI competition dataset 2A.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	Std	<i>p</i> value
Method 2	90.95	67.29	94.52	63.87	64.48	70.33	70.29	97.95	95.00	79.41	13.85	
Reference [43]	91.49	60.56	94.16	76.72	58.52	68.52	78.57	97.01	93.85	79.93	14.13	0.95
Reference [8]	88.89	59.03	90.28	78.47	62.50	75.00	72.92	93.06	87.50	78.63	11.63	0.90
Reference [44]	88.89	51.39	96.53	70.14	54.86	71.53	81.25	93.75	93.75	78.01	16.04	0.85
Reference [45]	90.28	54.17	93.75	64.58	57.64	65.28	62.50	90.97	85.42	73.84	15.02	0.45

TABLE 6: Kappa value comparison with other published results.

Subject	A01	A02	A03	A04	A05	A06	A07	A08	A09	Mean	<i>p</i> value
Method 2	0.77	0.38	0.76	0.47	0.27	0.42	0.73	0.81	0.85	0.607	
Method 1	0.66	0.39	0.78	0.47	0.25	0.41	0.72	0.79	0.83	0.589	0.8632
Reference [43]	0.86	0.24	0.70	0.68	0.36	0.34	0.66	0.75	0.82	0.60	0.9586
TSLDA [10]	0.74	0.38	0.72	0.50	0.26	0.34	0.69	0.71	0.76	0.567	0.6894
Winner 1 [46]	0.68	0.42	0.75	0.48	0.40	0.27	0.77	0.75	0.61	0.57	0.7051
Reference [8]	0.71	0.46	0.76	0.44	0.26	0.37	0.79	0.75	0.61	0.57	0.7245
Reference [14]	0.75	0.49	0.76	0.49	0.34	0.36	0.68	0.76	0.76	0.60	0.9353

TABLE 7: Five-folder cross validation by method 2 applied on BCI III dataset IIIa.

Subject	Method 2	Reference [47]	Reference [48]	Reference [49]	Reference [50]
k3b	91.67	90.00	86.67	94.20	94.44
k6b	75.00	76.25	81.67	69.00	62.50
l1b	81.67	77.91	85.00	78.60	78.33
Mean	82.78	81.38	84.44	80.60	78.42
<i>p</i> value		0.96	0.42	0.94	0.73

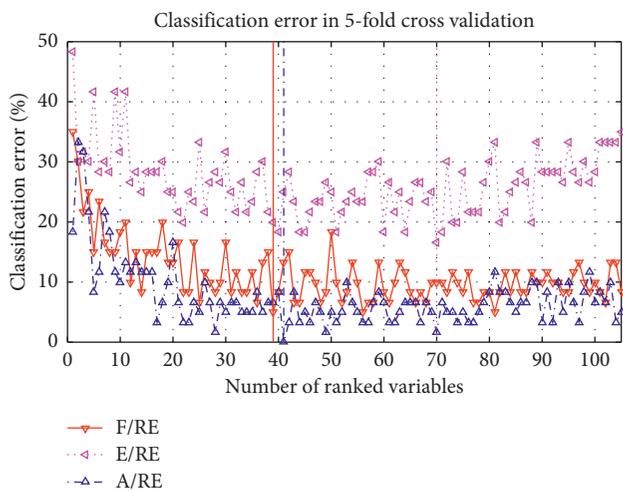


FIGURE 12: Classification accuracy corresponding to the number of selected variables of S01.

high ranking features to strengthen the expressive ability of the features.

6. Discussions

In this paper, we proposed a novel SSDT framework combined with classifiers to improve the performance of classifiers for multiclass MI tasks. We also proposed a novel NLDR method named SJGDA, and this NLDR method performs better than both *semi*-JMI and GDA on different datasets. In the following paragraphs, we have discussed the two methods in detail.

Method 1 indicates the drawback of FGMDRM, and then the novel SSDT framework is used to improve the accuracy for each individual. As shown in Table 2, compared with other published results, method 1 gets a quite good result in the case of processing the EEG signals of fixed frequency segment (8–30 Hz).

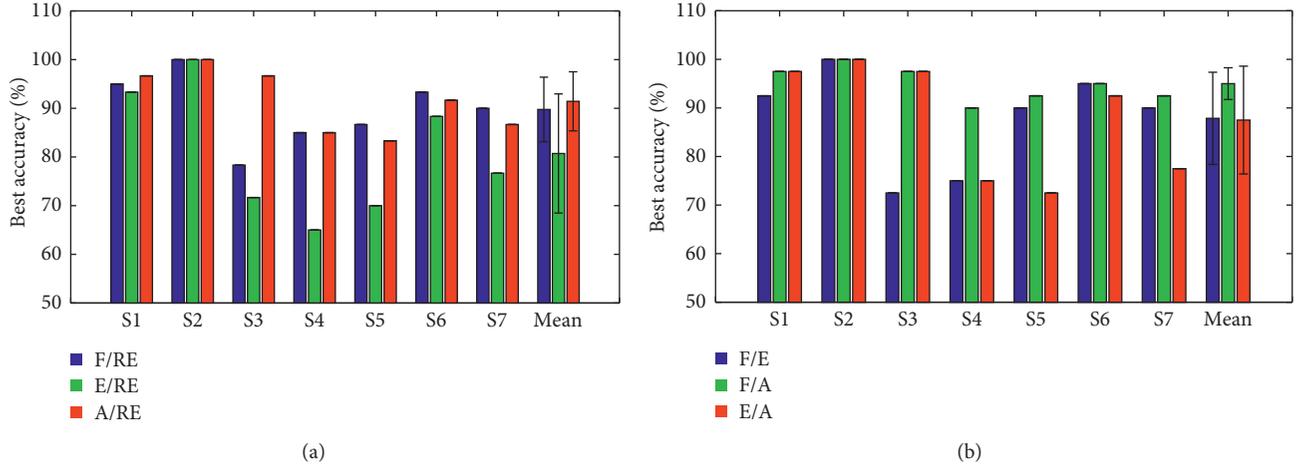


FIGURE 13: The comparison of three MI tasks in OVR scheme of seven subjects: (a) F/RE, E/RE, and A/RE; (b) F/E, F/A, and E/A.

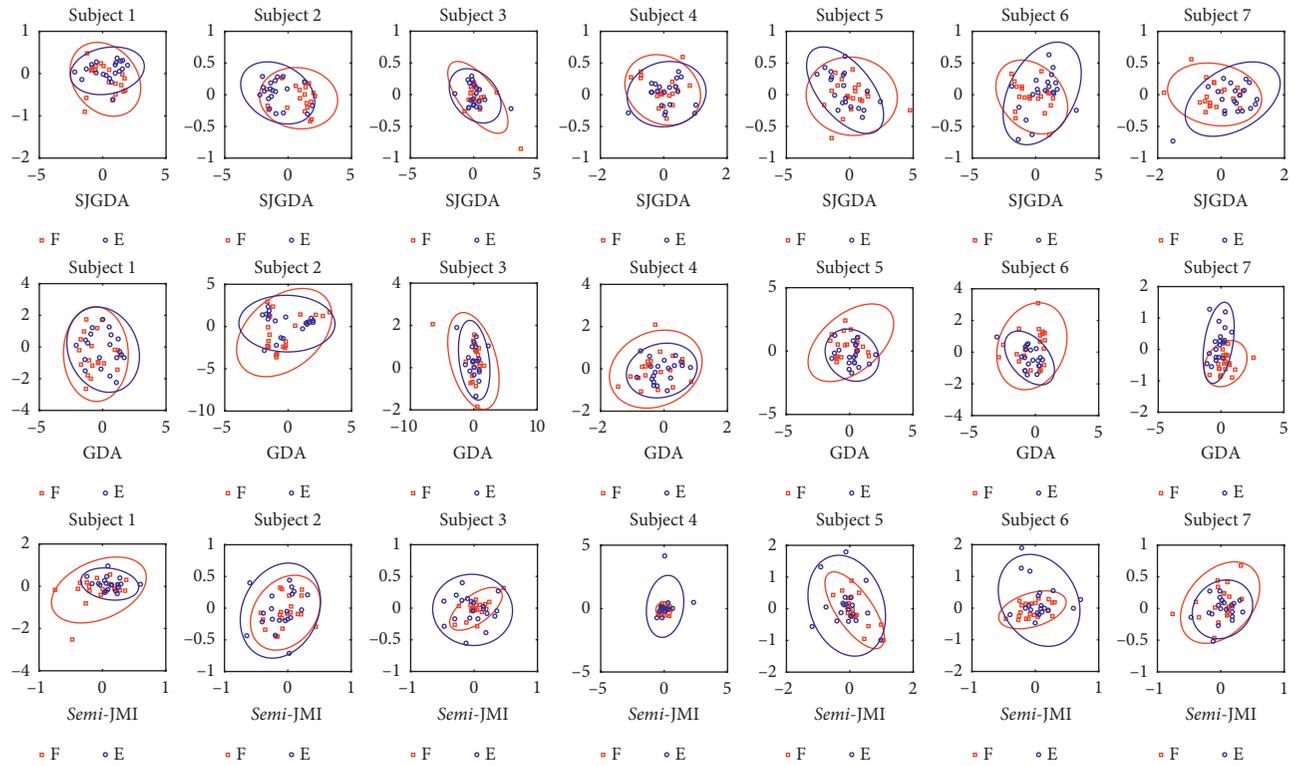


FIGURE 14: Feature distribution of F/E MI tasks extracted by SJGDA, GDA, and *semi*-JMI in dataset 2.

As shown in Table 6, Gaur et al. [43] proposed SS-MEMDBF to select the subject-specific frequency to obtain enhanced EEG signals which represent MI tasks related to μ and β rhythms, then classification with the Riemannian distance directly. TSLDA was proposed by Barachant et al. [10], and the covariance matrices are mapped onto a higher dimensional space where they can be vectorized and treated as Euclidean objects. Ang et al. [46] is the winner of the competition, FBCSP and multiple OVR classifiers were used for MI tasks, and achieved the mean kappa value of 0.57. Sadatnejad and Shiry Ghidary [8] proposed a new kernel for NLDR over the manifold of SPD matrices, the

kappa value is 0.576. Davoudi et al. [14] considered the geometry of SPD matrices and provides a low-dimensional representation of the manifold with high-class discrimination, and the best result of this method in terms of the kappa value is 0.60.

In method 2, SJGDA is used to get more obvious vectors from the tangent vectors, and a SSDT-KNN classifier is used to identify different MI tasks. Combined with SJGDA and SSDT-KNN, we have achieved a better performance compared with method 1 ($p = 0.8632$), Reference [43] ($p = 0.9586$), TSLDA ($p = 0.6894$), winner 1 ($p = 0.7051$), Reference [8] ($p = 0.7245$), and Reference [14] ($p = 0.9353$).

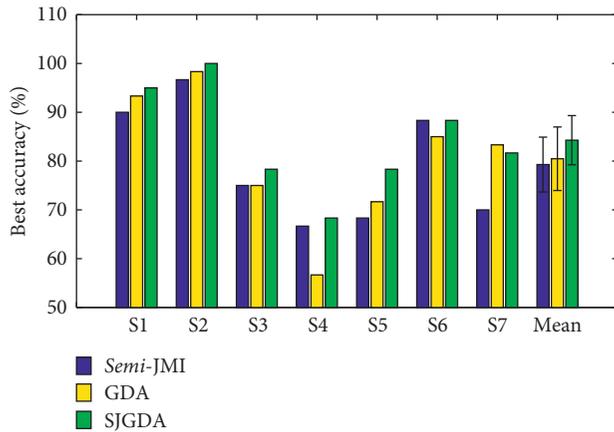


FIGURE 15: The comparison of the three different feature selection methods (*semi*-JMI, GDA, and SJGDA) with the same classifier (DT-KNN) of three MI tasks classification accuracy for each subject is depicted. The results are obtained by 5-folder cross validation.

It is clear that the proposed method in this paper is effective for MI tasks in a BCI system.

In order to prove the effectiveness of the proposed method 2, we tested it on two other datasets. As shown in Table 7 and Figure 15, method 2 achieves good classification results on two datasets.

7. Conclusion

The experimental results of method 1 show that the proposed classification framework significantly improves the classification performance of the classifier. The experimental results of method 2 show that the SJGDA algorithm proposed in this paper is superior to GDA and *semi*-JMI in feature extraction, and method 2 has the highest recognition rate in this paper. However, as the classifiers in the SSDT framework is substitutable, the focus of the next work is to combine more advanced classifiers with SSDT to increase the recognition rate of the BCI systems.

Data Availability

The dataset 1 and dataset 2 used to support the findings of this study are available from <http://bnci-horizon-2020.eu/database/data-sets>. The dataset 3 used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

Northeast Electric Power University (Grant number BSJXM-201521) and Jilin City Science and Technology Bureau (Grant number 20166012).

References

- [1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.
- [2] N. Birbaumer, "Breaking the silence: brain-computer interfaces (BCI) for communication and motor control," *Psychophysiology*, vol. 43, no. 6, pp. 517–532, 2006.
- [3] K. K. Ang, C. Guan, K. S. Phua et al., "Transcranial direct current stimulation and EEG-based motor imagery BCI for upper limb stroke rehabilitation," in *Proceedings of 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 4128–4131, IEEE, San Diego, CA, USA, August 2012.
- [4] B. Xu, S. Peng, A. Song, R. Yang, and L. Pan, "Robot-aided upper-limb rehabilitation based on motor imagery EEG," *International Journal of Advanced Robotic Systems*, vol. 8, no. 4, p. 40, 2011.
- [5] F. Wang, X. Zhang, R. Fu, and G. Sun, "Study of the home-auxiliary robot based on BCI," *Sensors*, vol. 18, no. 6, p. 1779, 2018.
- [6] F. Wang, H. Wang, and R. Fu, "Real-Time ECG-based detection of fatigue driving using sample entropy," *Entropy*, vol. 20, no. 3, 2018.
- [7] Z. J. Koles, M. S. Lazar, and S. Z. Zhou, "Spatial patterns underlying population differences in the background EEG," *Brain Topography*, vol. 2, no. 4, pp. 275–284, 1990.
- [8] K. Sadatnejad and S. Shiry Ghidary, "Kernel learning over the manifold of symmetric positive definite matrices for dimensionality reduction in a BCI application," *Neurocomputing*, vol. 179, pp. 152–160, 2016.
- [9] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review," *Brain-Computer Interfaces*, vol. 4, no. 3, pp. 155–174, 2017.
- [10] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain-computer interface classification by riemannian geometry," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, 2012.
- [11] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Common spatial pattern revisited by riemannian geometry," in *Proceedings of 2010 IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 472–476, IEEE, Saint Malo, France, October 2010.
- [12] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "A brain-switch using riemannian geometry," in *Proceedings of 5th International Brain-Computer Interface Conference 2011 (BCI 2011)*, pp. 64–67, Graz, Austria, September 2011.
- [13] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Classification of covariance matrices using a riemannian-based kernel for BCI applications," *Neurocomputing*, vol. 112, pp. 172–178, 2013.
- [14] A. Davoudi, S. S. Ghidary, and K. Sadatnejad, "Dimensionality reduction based on distance preservation to local mean (DPLM) for spd matrices and its application in BCI," 2016, <https://arxiv.org/abs/1608.00514>.
- [15] S. Brandl, K.-R. Müller, and W. Samek, "Robust common spatial patterns based on Bhattacharyya distance and gamma divergence," in *Proceedings of 2015 3rd International Winter Conference on Brain-Computer Interface (BCI)*, pp. 1–4, IEEE, Gangwon-Do, South Korea, January 2015.
- [16] W. Samek, M. Kawanabe, and K.-R. Müller, "Divergence-based framework for common spatial patterns algorithms,"

- IEEE Reviews in Biomedical Engineering*, vol. 7, pp. 50–72, 2014.
- [17] W. Samek and K.-R. Müller, “Information geometry meets BCI spatial filtering using divergences,” in *Proceedings of 2014 International Winter Workshop on Brain-Computer Interface (BCI)*, pp. 1–4, IEEE, Gangwon, South Korea, February 2014.
- [18] W. Samek and M. Kawanabe, “Robust common spatial patterns by minimum divergence covariance estimator,” in *Proceedings of 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2040–2043, IEEE, Florence, Italy, May 2014.
- [19] S. Kumar, K. Mamun, and A. Sharma, “CSP-TSM: optimizing the performance of riemannian tangent space mapping using common spatial pattern for mi-BCI,” *Computers in biology and medicine*, vol. 91, pp. 231–242, 2017.
- [20] C. Lindig-León, N. Gayraud, L. Bougrain, and M. Clerc, “Comparison of hierarchical and non-hierarchical classification for motor imagery based BCI systems,” in *Proceedings of The Sixth International Brain-Computer Interfaces Meeting*, Pacific Grove, CA, USA, May-June 2016.
- [21] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [22] O. Kramer and D. Lückehe, “Visualization of evolutionary runs with isometric mapping,” in *Proceedings of 2015 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1359–1363, IEEE, Sendai, Japan, May 2015.
- [23] K. Q. Weinberger, F. Sha, and L. K. Saul, “Learning a kernel matrix for nonlinear dimensionality reduction,” in *Proceedings of the Twenty-First International Conference on Machine Learning*, p. 106, ACM, Banff, Canada, July 2004.
- [24] L. Van Der Maaten, “Learning a parametric embedding by preserving local structure,” *RBM*, vol. 500, p. 26, 2009.
- [25] L. Van Der Maaten, “Accelerating t-SNE using tree-based algorithms,” *Journal of Machine Learning Research*, vol. 15, pp. 3221–3245, 2014.
- [26] D. Lee, S.-H. Park, and S.-G. Lee, “Improving the accuracy and training speed of motor imagery brain-computer interfaces using wavelet-based combined feature vectors and Gaussian mixture model-supervectors,” *Sensors*, vol. 17, no. 10, p. 2282, 2017.
- [27] X. Xie, Z. L. Yu, H. Lu, Z. Gu, and Y. Li, “Motor imagery classification based on bilinear sub-manifold learning of symmetric positive-definite matrices,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 6, pp. 504–516, 2017.
- [28] M. T. Harandi, M. Salzmann, and R. Hartley, “From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices,” in *Proceedings of European Conference on Computer Vision*, pp. 17–32, Springer, Zurich, Switzerland, September 2014.
- [29] M. Moakher, “A differential geometric approach to the geometric mean of symmetric positive-definite matrices,” *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 3, pp. 735–747, 2005.
- [30] P. T. Fletcher and S. Joshi, “Principal geodesic analysis on symmetric spaces: statistics of diffusion tensors,” in *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, M. Sonka, I. A. Kakadiaris, and J. Kybic, Eds., pp. 87–98, Springer, Berlin, Heidelberg, 2004.
- [31] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, “Riemannian geometry applied to BCI classification,” in *Proceedings of International Conference on Latent Variable Analysis and Signal Separation*, pp. 629–636, Springer, St. Malo, France, September 2010.
- [32] H. Guo and S. B. Gelfand, “Classification trees with neural network feature extraction,” *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 923–933, 1992.
- [33] S. R. Safavian and D. Landgrebe, “A survey of decision tree classifier methodology,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.
- [34] Y.-H. Shao, W.-J. Chen, W.-B. Huang, Z.-M. Yang, and N.-Y. Deng, “The best separating decision tree twin support vector machine for multi-class classification,” *Procedia Computer Science*, vol. 17, pp. 1032–1038, 2013.
- [35] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján, “Conditional likelihood maximisation: a unifying framework for information theoretic feature selection,” *Journal of Machine Learning Research*, vol. 13, pp. 27–66, 2012.
- [36] K. Sechidis and G. Brown, “Simple strategies for semi-supervised feature selection,” *Machine Learning*, vol. 107, no. 2, pp. 357–395, 2017.
- [37] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, “A unifying view on dataset shift in classification,” *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012.
- [38] G. Baudat and F. Anouar, “Generalized discriminant analysis using a kernel approach,” *Neural Computation*, vol. 12, no. 10, pp. 2385–2404, 2000.
- [39] M. Haghghat, S. Zonouz, and M. Abdel-Mottaleb, “Cloudid: trustworthy cloud-based and cross-enterprise biometric identification,” *Expert Systems with Applications*, vol. 42, no. 21, pp. 7905–7916, 2015.
- [40] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer Science & Business Media, Berlin, Germany, 2013.
- [41] C. Brunner, R. Leeb, G. Müller-utz, A. Schlögl, and G. Pfurtscheller, BCI Competition 2008–Graz Data Set A, vol. 16, Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces) Graz University of Technology, Graz, Austria, 2008.
- [42] B. Blankertz, K. R. Müller, D. J. Krusienski et al., “The BCI competition III: validating alternative approaches to actual BCI problems,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 153–159, 2006.
- [43] P. Gaur, R. B. Pachori, H. Wang, and G. Prasad, “A multi-class EEG-based BCI classification using multivariate empirical mode decomposition based filtering and riemannian geometry,” *Expert Systems with Applications*, vol. 95, pp. 201–211, 2018.
- [44] F. Lotte and C. Cuntai Guan, “Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 355–362, 2011.
- [45] H. Raza, H. Cecotti, Y. Li, and G. Prasad, “Adaptive learning with covariate shift-detection for motor imagery-based brain-computer interface,” *Soft Computing*, vol. 20, no. 8, pp. 3085–3096, 2015.
- [46] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, “Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b,” *Frontiers in Neuroscience*, vol. 6, p. 39, 2012.
- [47] H. Baali, A. Khorshidtalab, M. Mesbah, and M. J. E. Salami, “A transform-based feature extraction approach for motor imagery tasks classification,” *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 3, pp. 1–8, 2015.
- [48] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, “Characterization of four-class motor imagery EEG data for the

- BCI-competition 2005,” *Journal of Neural Engineering*, vol. 2, no. 4, pp. L14–L22, 2005.
- [49] M. Grosse-Wentrup and M. Buss, “Multiclass common spatial patterns and information theoretic feature extraction,” *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 8, pp. 1991–2000, 2008.
- [50] I. Koprinska, “Feature selection for brain-computer interfaces,” in *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 106–117, Springer, Bangkok, Thailand, April 2009.

Research Article

Geoffrey: An Automated Schedule System on a Social Robot for the Intellectually Challenged

Edmanuel Cruz,^{1,2} Félix Escalona ,¹ Zuria Bauer,¹ Miguel Cazorla ,¹ José García-Rodríguez ,¹ Ester Martínez-Martin ,¹ José Carlos Rangel,^{1,2} and Francisco Gomez-Donoso ¹

¹Institute for Computer Research, University of Alicante, P. O. Box 99. 03080, Alicante, Spain

²RobotSIS, Universidad Tecnológica de Panamá, Panama City, Panama

Correspondence should be addressed to Francisco Gomez-Donoso; fgomez@dccia.ua.es

Received 11 October 2018; Accepted 15 November 2018; Published 2 December 2018

Academic Editor: Michele Migliore

Copyright © 2018 Edmanuel Cruz et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The accelerated growth of the percentage of elder people and persons with brain injury-related conditions and who are intellectually challenged are some of the main concerns of the developed countries. These persons often require special cares and even almost permanent overseers that help them to carry out diary tasks. With this issue in mind, we propose an automated schedule system which is deployed on a social robot. The robot keeps track of the tasks that the patient has to fulfill in a diary basis. When a task is triggered, the robot guides the patient through its completion. The system is also able to detect if the steps are being properly carried out or not, issuing alerts in that case. To do so, an ensemble of deep learning techniques is used. The schedule is customizable by the carers and authorized relatives. Our system could enhance the quality of life of the patients and improve their self-autonomy. The experimentation, which was supervised by the ADACEA foundation, validates the achievement of these goals.

1. Introduction

The increment of life expectancy and the low mortality rates in developed countries are bringing an accelerated growth of the percentage of elder people. As remarked by the United Nations (<https://population.un.org/ProfilesOfAgeing2017/index.html>), currently, 12.74% of the population is above the threshold of 65-year-old, but as for 2050, this is expected to grow up to 27.04% of the global population. This quick population aging is one of the principal concerns of developed countries and one of the priority lines of research. In addition to the elder people, thousands of people worldwide are affected by brain-related injuries nowadays. These diseases can be caused by different situations such as trauma, accident, or even by genetic affections. The alterations that can arise after an acquired brain injury with increasing age include the loss of intellectual abilities of different severity. These disabilities interfere with social or occupational

functioning, memory, or abstract thinking disorders, the inability to find similarities and differences between related words, or the difficulty to perform common domestic tasks, among others. Specifically, one of the most common conditions is the omission of the natural order to perform a task. For instance, the affected people for this condition would brush their teeth before applying the toothpaste or take the toothbrush but eventually forget what comes next.

One of the worst outcomes of being part of these collectives is the reduced personal autonomy. Elder and intellectually challenged people often require the special attention of a therapist that helps them to perform diary tasks such as tying the shoelaces, taking a shower, or having the meals.

In this context, we propose the implementation of an automated schedule system on a social robot that would assist the patients in their daily tasks at home. The system will notify the programmed tasks to the patients on a

scheduled time and will also help to achieve the actions by walking the patient through them. It will detect if the user is doing well and give feedback if not.

The implementation of the proposed system would improve the personal autonomy and enhance the quality of life of elderly and intellectually challenged people.

So, the main contributions of this work are

- (i) A programmable schedule system deployed on a social robot
- (ii) An integration of different methods in order to monitor if the patient is performing the actions he is intended to perform
- (iii) A navigation system which consists of a mix of semantic localization methods and the traditional SLAM

The rest of the paper is structured as follows. First, the state of the art is reviewed in Section 2. In Section 3, our proposal is thoroughly explained by giving an overview and then focusing on each piece of the system. Then, the experimentation is detailed in Section 4. Finally, the conclusions and limitations of this work are discussed in Section 5.

2. Related Works

The growing number of elder people is being increasingly important to promote the role and technological advance of social and assistive robotics. Speaking about Social Robotics creates the necessity to actually define what a social robot is. According to [1], it is defined as “*A physical entity embodied in a complex, dynamic, and social environment sufficiently empowered to behave in a manner conducive to its own goals and those of its community.*”

In this specific field, we have to divide the different works based on their application. There are projects for assistance in medical environments [2, 3], whilst some others are focused on the emotional and cognitive tasks [4, 5], and there also are projects for a social assistive task in different environments. This state of art is focused on this last issue.

In 1998, PAM-AID (Personal Adaptive Mobility Aid) [6] was created. This system aims to provide both physical support during walking and obstacle avoidance. It used sonar, infrared proximity sensors, and bumpers switches to get information from the environment.

One-year later, the system proposed in [7] emerged. It described the implementation of a control architecture for robots designed to combine a manipulation task with a motion controller that used the operational space formulation to define and implement arm trajectories and object manipulation.

The same year appeared the first so-called intelligent wheelchair [8]. The system provided different functions: from fully autonomous navigation in an unknown crowded environment to partially autonomous local maneuvers. Two years later, on the same topic, the study [9] was created, which described the mounting of a robotic arm to a powered wheelchair to assist disabled users in daily activities.

In 2003, three different types of work appeared. First, Falcone et al. [10] describe the efforts to design, prototype, and test a low-cost, highly competent personal rover for the domestic environment. Then, Pineau et al. [11] describe a mobile robotic assistant developed to assist elderly individuals with mild cognitive and physical impairments, as well as support nurses in their daily activities. They used three software modules: an automated reminder system, people tracking and detection system, and a high-level robot controller. Finally, Pollack et al. [12] use AI techniques to model an individual’s daily plans, observe and reason about the execution, and make decisions about whether and when it is most appropriate to issue reminders.

Years later, it appeared the PAMM [13] project, which is a system for support and guidance. The PAMM detects and maneuvers away from obstacles, and it uses an upward looking camera for localization and also can communicate with a central computer. The central computer provides the system with a map of the facility including the location. In turn, the system provides the central computer with the user’s location, health status, and requests.

In 2010, the system described in [14] was proposed. It aims at designing a socially assistive robot to monitor the performance of the user during a seated arm exercise scenario, and the main purpose was to provide motivation to the user to complete the task and to improve performance.

Also this year, “the home exploring robotic butler” (HERB) [15] was published. It can efficiently perform mapping tasks, searching, and navigation through indoor environments, recognize and localize several common household objects, and perform complex manipulation tasks.

In 2011, the system described in [16] appeared. It was an indoor mobile robot for taking care of the elderly. It has a human physiological parameters monitor system, which can take care up to six nursed persons by using a variety of sensors.

The ASIBOT [17] was published one-year later. It helps users to perform a variety of tasks in common living environments. The robot is able to autonomously climb from one surface to another, fixing itself to the best place to perform each task. It also can be attached to a wheelchair, giving the user the possibility to move along with it as a bundle.

A new iteration of the aforementioned HERB [15] system emerged in 2012. The HERB 2.0 [18] consists of a two-handed mobile manipulator that can perform useful tasks for and with people in human environments.

In 2014, it was created as a multiuser human-robot interaction (HRI) [19] system architecture to allow the social robot Tangy to autonomously plan, schedule, and facilitate multiuser activities that consider the users’ necessities. During the activities, the robot was able to interact with a group of users providing group-based and individualized assistance based on the needs of the individual. The same year, the robotic nursing assistant (RoNA) was created [20], which would assist nurses while performing intensive tasks and prevent musculoskeletal injuries among health care workers.

The robotic exercise tutor [21] was published in 2017. The created humanoid robot learns exercise routines from a human trainer and performs them in front of the elderly. Its main task was to monitor the performance of the patients and provide feedback.

In 2018, deep learning (DL) algorithms were introduced into assistive robot systems. For instance, a remote health care system based on moving robots intended for the elderly at home was proposed in [22]. The robot is able to perform different kind of tasks. The user can control the robot and call it using voice commands or with a phone, and it also performs object detection and pose estimation. It even can monitor the posture of the elderly and collect and transmit the data recorded by a set of sensors connected to the robot to the cloud for further analysis. On its behalf, PHAROS, a robotic assistant for assisting elderly in their daily physical activities at home, was proposed in [23]. This interactive robot platform is divided into two modules: the recommender (recommends activities at a scheduled time) and the human exercise recogniser (as its name implies, it is the identifier of the human pose). This system works in real time and uses deep learning methods to properly recognize the performed physical exercises.

3. Proposal

In this work, we propose a robotic system for monitoring and interacting with people affected by cognitive disabilities and elder people. The system will guide the patient through their daily tasks helping, guiding, and encouraging them to follow a preset schedule. The ultimate goal of the proposed system is to improve the quality of life of dependent people and their self autonomy.

The system is composed of a programmable schedule and a contextual schedule. The programmable schedule is composed of a list of tasks that the patient must perform at a certain time. This kind of tasks could be delayed if the patient is authorized to do so. On the contrary, the contextual schedule is composed of a list of tasks that the patient is authorized to perform on demand, and it will depend on the room the user is located.

As shown in Figure 1, when the patient is notified to complete a task of the programmable schedule, the patient could try to complete it now or delay it. If the patient chooses to perform it upon requirement, the robot will first guide the patient to the appropriate room, and then, it will provide instructions in order to perform the required task. If the patient delayed a task, the system will ask him to complete it later.

Whether the tasks come from the programmed or the contextual schedule, each one is composed of a list of actions, which are goals the patient must accomplish to complete a task. The tasks are assigned and setup by the therapist in charge of the patient or authorized relatives. Currently, our system comprehends four main different types of actions: object recognition, behavior recognition, QR recognition, and spend time action. As aforementioned, the person in charge of the patient can combine these actions to construct high-level tasks. For instance, the task “pour a

glass of water” would be composed of three different actions. First, the robot would ask the patient to get a glass and show it, and then, it would ask to show the bottle of water. Both actions would use the object recognition engine in order to detect a glass and a bottle. Finally, the behavior recognition engine would be setup to detect the actual action of pouring. If the user consistently fails to complete an action, the task is automatically aborted (and delayed if it was configured like so). In addition, the carers could be notified of this event if they choose to. Similarly, the task “take the medicine A” would involve two tasks. First, the robot would ask to show the medicine A package, which would be labeled with a QR code. The action in this case would use the QR recognition engine to check if the object showed by the patient is correct. Then, it would spend some time idle waiting for the patient to take the medicines by making use of a spend time action. Through the achievement of the tasks, the robot is continually providing both visual and speech feedback to inform and encourage the patient upon the completion of the task.

It is worth noting that the system is intended to be deployed in a social robot that will follow the patient wherever he goes.

3.1. The Social Robot. A Pepper robot was chosen in order to develop the proposed system. As shown in Figure 2, the Pepper robot is a human-shaped robot manufactured by Softbank Robotics. It features a variety of sensors such as radar, laser, RGB-D cameras, and microphones among others.

We adopted it for two main reasons: first, its appearance is familiar and engaging so the users feel comfortable when interacting with it. On the contrary, its features suit perfectly the requirements of our proposal. The requirements are a mobile base and size that allow the robot to easily move in indoor environments in order to guide the patient to the desired room and to follow him and a microphone that enables speech and voice recognition capabilities. They provide a natural interaction mechanism; a tablet to display relevant information and feedback and to offer an alternative interaction method; a camera to monitor the patient and detect if the given directions are being followed; and a front laser which is in charge of detecting obstacles and provide local localization capabilities.

It is worth noting that any other robot that meets the aforementioned requirements can be used to deploy the proposed system.

3.2. Object Recognition Engine. As aforementioned, there exist actions in the tasks that are about detecting a certain object, for instance, the detection of a glass in the task, “pour a glass of water.”

To do so, we implemented an object recognition engine (ORE). The ORE is based on the InceptionResNetV2 [24] architecture. This DL-based method is proven to provide one of the lowest Top-1 and Top-5 errors on the ImageNet ILSVRC 2012 [25] challenge. This architecture takes advantage of the inception concept and residual connections in

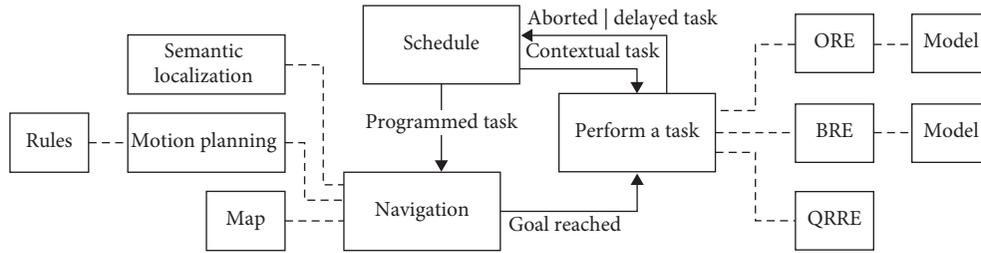


FIGURE 1: Diagram of the proposal.

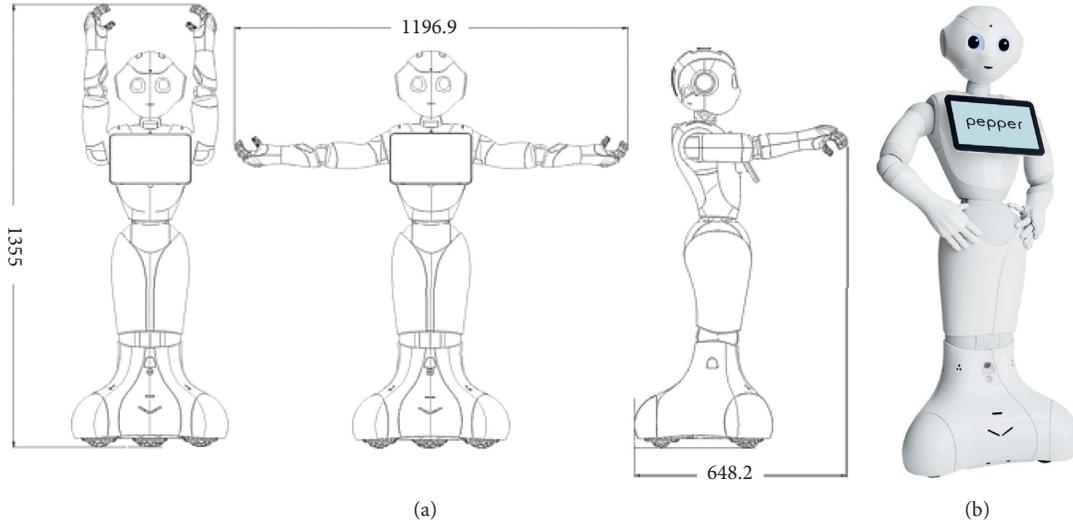


FIGURE 2: The proposed system is intended to be deployed on a Pepper robot, but it can be deployed in any robot that meets the requirements of our system. (a) Some physical features of the Pepper robot, whilst the (b) depicts its visual appearance.

order to obtain high accuracy rates while maintaining the computational cost at bay.

We adopted a classic convolutional neural network (CNN) scheme over region-based convolutional neural network (R-CNN) architectures to avoid false detections. For instance, when the robot asks for a glass, it is intended that the patient looked for the object and held it in front of the robot. Nonetheless, if the patient is in the kitchen, it is likely that there were glasses on the counter or on a table that a R-CNN would detect. Given this case, it is difficult to differentiate whether the patient is holding the object or it appeared in the background of the scene. On the contrary, taking advantage of a classic CNN approach, the object is correctly detected if only the object is depicted in the input image. This way, we force the patient to look for the desired object and to show it on purpose to the robot.

When the ORE is used by the system, it first captures an image using the camera of the robot. The image is classified by the ORE and if the returned label matches the desired object, the action is considered fulfilled. If not, the process is repeated up to a preestablished number of trials. If the object is not correctly detected, the whole task is aborted. We consider detection if any of the scores assigned to a label is above a certain threshold. If no label is above this threshold, no object is considered as detected.

Note that the ORE must be used when high generalization capabilities are required. As aforementioned, when asking for a glass, any glass would work. So, we need the classifier to recognize any kind of glass.

3.3. Behavior Recognition Engine. Once the necessary objects to carry out the requested action are identified, the next step is to properly recognize the user's behavior. For that, the first step is to robustly detect the person(s) within the image. However, this is not a straightforward task due to the required generality of the system. So, the designed behavior recognition engine (BRE) should be able to properly recognize the user's behaviors in different rooms and in different houses. Therefore, background subtraction techniques are discarded. In addition, no requirements about patient's appearance can be established. As a consequence, an abstraction mechanism is required. In particular, the skeleton-based representation Openpose [26, 27, 28] is used. Basically, this two-branch multistage convolutional neural network (CNN) outputs a 18-keypoint body skeleton for all the people in the image, independent on the background or the person, as illustrated in Figure 3.

From this body keypoint information, a new image focused on the human skeletons is generated. In this way, the

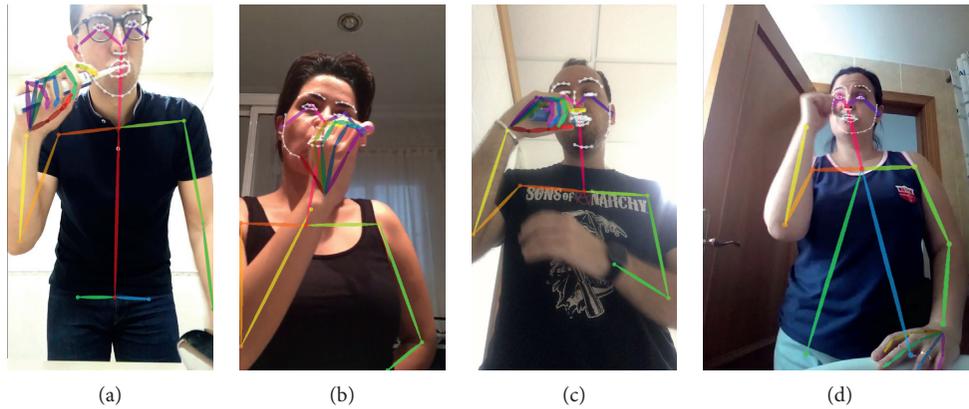


FIGURE 3: Human skeleton detection of several people in different bathrooms by using Openpose [26, 27, 28].

behavior recognition is reduced to a human pose classification. That is, each behavior can be defined as a sequence of several human poses such that their classification allows the system to recognize the represented behavior (see Figure 4 for example).

As a classification problem, a CNN architecture could fit. Particularly, in this paper, a *ResNet50* [29] was used. *ResNet50* is a deep residual network of 50 layers that have been trained to the task at hand.

Thus, the BRE flowchart can be described as follows: the robot RGB camera takes an image, from which a $224 \times 224 \times 3$ image with human skeletons is generated. This skeleton-based image feeds the pretrained *ResNet50* that outputs the observed behavior (Figure 5).

3.4. QR Recognition Engine. There are actions in the tasks that require a much finer detection, meaning, a certain instance of an object. For example, in the action “show the medicine A,” the robot is asking for a particular medicine. The ORE cannot be used in this case because the requested object is very specific, and its main goal is to provide high generalization capabilities. So, the QR recognition engine (QRRE) is intended to be used when the required specificity of the requested object is critical.

To do so, we implemented a QR code detector. The QRRE is based on Zbar [30], which is an open-source barcode and QR codes scanner. As expected, the objects must be manually tagged with the correspondent QR code, so the QRRE could be used to recognize them. We chose this method over the traditional object recognition pipeline [31, 32, 33, 34] because it is much faster and reliable.

When the QRRE is used by the system, first it captures an image using the camera of the robot. The image is fed to the QRRE, and if the returned label matches the desired object, the action is considered fulfilled. If no object is correctly detected within the preestablished number of trials, the whole task is aborted.

3.5. Speech Recognition Capabilities. In order to allow a natural way of interaction with the robot, our system takes advantage of the built-in speech recognition capabilities

offered by the Pepper robot. The speech recognition engine is provided by Nuance [35], which is a company experienced in this area. This company is in charge of developing top-tier commercial speech recognition software.

The speech recognition engine is able to identify predefined words and statements configured by the user. We adopted it as one of the main interaction methods our system offers and is used when the robot asks a question. For instance, the robot expects a “yes” or “ok” or a “no” when it asks the user if he would like to perform a programmed task now. The tablet of the robot displays a “listening” message, and the robot makes an acoustic alert whenever the speech recognition engine is expecting an answer from the patient, so he would know that he can interact with the robot using this method.

3.6. Semantic Localization System. The aim of the semantic localization system (SLS) is to compute the location at the semantic level. To do this, the work based the SLS on was presented [36]. In this work, an optimal methodology for a mobile robot to adapt its knowledge to new environments was proposed. This module works in the following way.

First, the robot captures images of the environment and tries to classify them using an initial pretrained model. The images used to train this model come from unknown and different homes. As it is deployed in a new environment, it is likely that the system obtains low accuracy rates. This is due to the different visual features of the new environment and the environments in which the model was trained in the first place. In this case, we can provide information to the robot to collect data and reidentify the locations. In case the category provided by the user is not considered so far by the model, it will be added as a new category. This way, the robot can easily increase and fit its knowledge to the new environment.

It is worth noting that the model fitting of the SLS is performed before the robot is actually deployed, so it can precisely localize itself once deployed without the interference of the patient.

To achieve this goal, we use the architecture showed in Figure 6. This works as follows: an input image is forwarded

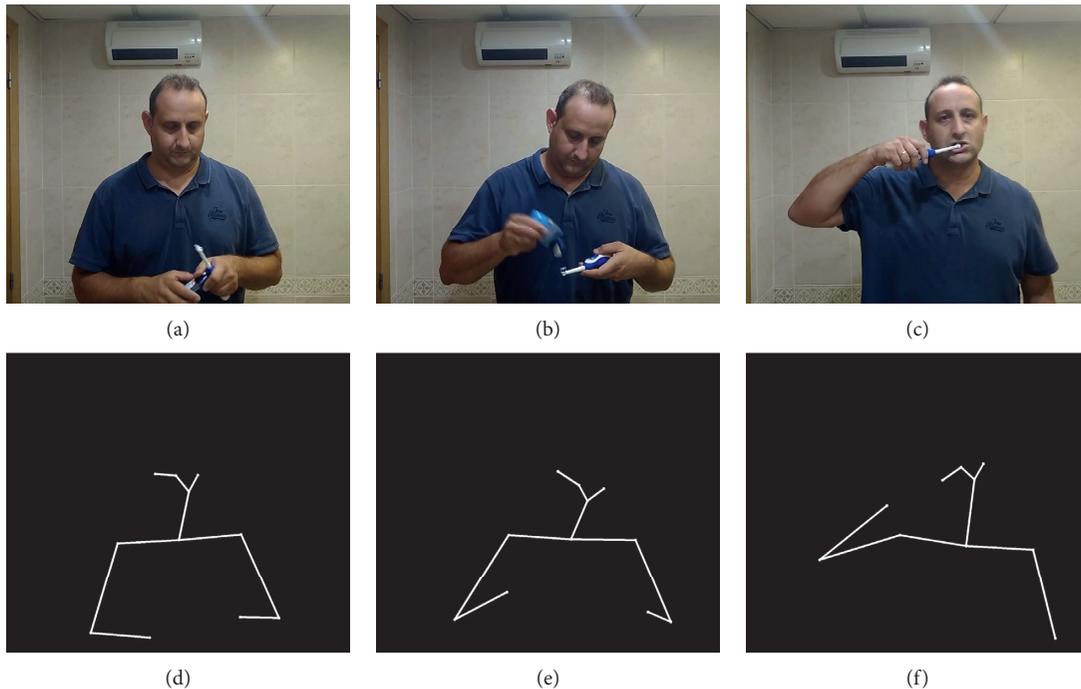


FIGURE 4: Human poses representing the brushing teeth behavior. (a)–(c) depicts images as captured by the robot, whilst (d)–(f) shows the corresponding estimated skeleton.

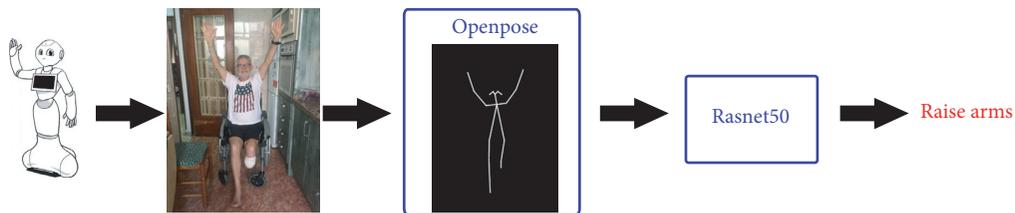


FIGURE 5: Flowchart corresponding to the implemented behavior recognition engine (BRE): the top Pepper’s RGB camera captures an image that is processed by Openpose to get a skeleton-focused image. This image feeds the trained ResNet50 to properly recognize the observed behavior.

to the ResLoc CNN architecture. This is a classic CNN architecture which lasts until the fully convolutional layer was removed, so the output is the visual features descriptor for the input image. As a result, the output of ResLoc CNN is a 2,048 dimensions feature vector.

The visual features and the correspondent categories of each image of the training dataset are extracted using the ResLoc CNN part of the architecture and inserted on the features database. This features database is a model that stores the learned data, which are the features of the training samples. This model is used during the inference stage.

On the inference stage, the unknown image is forwarded to ResLoc CNN in order to extract the visual feature vector. Then, a K -nearest neighbors (KNN) classifier performs a query on the feature database using the recently computed feature vector. Next, a polling is carried out among the categories of the neighbors, and the most voted category is returned as the final classification of the unknown image.

The performance of the KNN is highly dependent on the k parameter (number of neighbors). Experimentation on

this matter is carried out to set the best performing k . We used the Annoy [37] implementation of the (approximate) KNN classifier.

Then, the model is specialized even more with samples that come for the actual house in which it is deployed. The new samples are inserted only if the localization fails in a certain room.

As a consequence of this method, the SLS is always updating its model to prevent losing performance, thus, adapting it as the time goes past. This is specially useful as the appearance of the environment is inevitably going to change. For instance, the furniture is eventually being changed or rearranged, the walls are being painted of another color, or the home appliances are being replaced.

3.7. Motion Planning System. When a new location goal is determined as a consequence of triggering a task that must be performed on another room, we need a system that calculates the path from the current room to the target. This

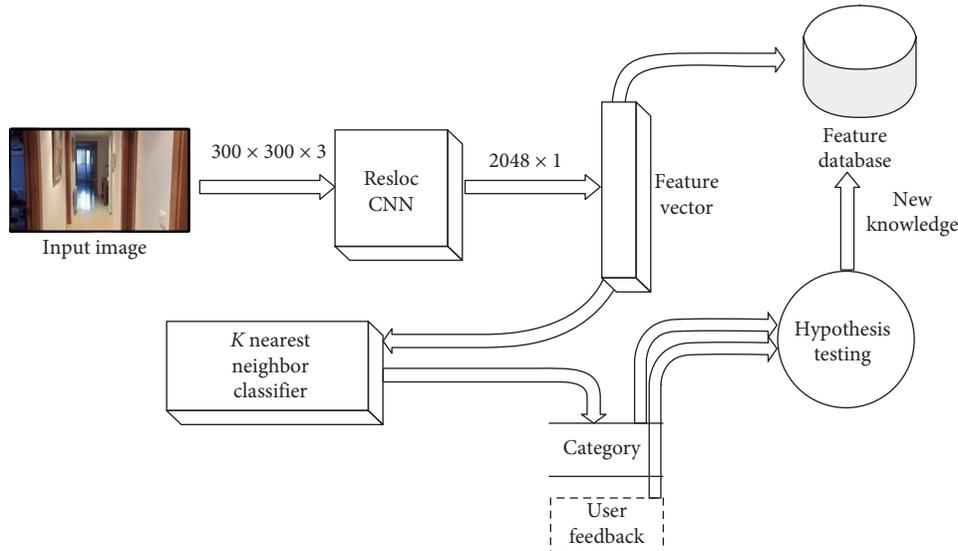


FIGURE 6: This architecture uses the features of a ResLoc CNN with a vector of 2,048 features as the output. The training samples are forwarded to the ResLoc CNN in order to extract their feature vectors. The feature vectors construct the model of a KNN classifier.

task can be done using a simplified map of the environment and an expert system that computes all the paths between the actual location room and the destination room using some connection rules and facts. We named this system the motion planning system (MPS).

The map is modeled as a graph where the nodes are rooms, doors, and intersections between location access, and the edges are the connection between these places. These connections have an associated direction of movement for the transition (north, east, south, and west) and a travel cost that represents the distance between the nodes. Also, we define the transition matrix between node types, where we represent the action that must be performed going from the node type A to B , as shown in Table 1. Cross type is a node outside a tagged room where various ways join. Interior type is a node inside a tagged room where various ways join. The actions are cross the door (cd), follow the corridor (fc), and ND (not defined).

The expert system that computes the paths between nodes has been developed in Prolog, a logic programming language that provides great tools for this kind of task, such as declarative rules and unification (for restrictions management) and backtracking (for graph exploration).

The knowledge is divided into two separated files: facts and rules. The facts are specific to the concrete environment that is being modeled. They contain the definition of the nodes with their types (rooms, doors, cross, and interiors), the connection between nodes with the direction and the associated movement cost, and a dynamic predicate that indicates if a door is closed, which can be modified at runtime with the information provided by the robot sensors.

The rules are common to every environment modeled in this way. They check the connectivity between the nodes (direct or indirect) and calculate the path, set of directions, set of actions, and the cost from node A to B . In facts files, the direct connectivity between nodes is represented only in one direction, so we have defined rules that allow the reverse

TABLE 1: Transition matrix between node types.

From/to	Room	Door	Cross	Interior
Room	cd	cd	ND	fc
Door	cd	fc	fc	cd
Cross	ND	fc	fc	ND
Interior	fc	cd	ND	fc

computation of this connectivity, looking for connections from node A to B and from node B to A reversing the direction of the movement, as shown in Sourcecode 1. The same principle has been applied to the computation of the actions, as shown in Sourcecode 2. The predicate `action/3` is the Prolog representation of the transition matrix shown in Table 1.

For the building of the paths between the nodes, we recursively search for those that are directly connected to the current one until we reach the final node. We have to notice that we are looking for paths without loops (trees), so we do not allow the repetition of any node in them. Without this restriction, the computation of this exploration would hang and enter an infinite loop.

Due to the flexibility of the Prolog module, we cannot only calculate the paths between the defined nodes A and B , but we can make much more queries, like discovering all the accessible nodes from every node, using the same facts and rules knowledge.

3.8. Navigation and Mapping. We also relied on the ROS framework for the complete control of the movement of the robot. This framework provides utilities to interact with the robot and the mapping and navigation methods we adopted for our system.

The task of moving the robot from its current location to another part of the house requires a list of waypoints, which correspond to the labels (one per room) that are used by the

```

(1) %Directions
(2) dir (north).
(3) dir (south).
(4) dir (east).
(5) dir (west).
(6) %Reversibility of orientations
(7) revDir (east, west).
(8) revDir (north, south).
(9) %Revert orientations
(10) isRevDir (X, Y):- revDir (X, Y).
(11) isRevDir (X, Y):- revDir (Y, X).
(12) % Look for direct connections
(13) hasConnection (X, X, none, 0).
(14) hasConnection (X, Y, Direction, Cost):- dir (Direction), connection (X, Y, Direction, Cost).
(15) hasConnection (X, Y, Direction, Cost):- dir (Direction), isRevDir (Direction, Reversed),
(16) connection (Y, X, Reversed, Cost).

```

SOURCECODE 1: Definition of connection rules.

```

(1) %Action definition
(2) action (X, X, none):- isPlace (X).
(3) action (X, Y, cd):- room (X), room (Y), X\ = Y.
(4) action (X, Y, fc):- cross (X), cross (Y), X\ = Y.
(5) action (X, Y, fc):- door (X), door (Y), X\ = Y.
(6) action (X, Y, fc):- interior (X), interior (Y), X\ = Y.
(7) action (X, Y, cd):- room (X), door (Y), not (closed (Y)), X\ = Y.
(8) action (X, Y, fc):- room (X), interior (Y), X\ = Y.
(9) action (X, Y, fc):- cross (X), door (Y), X\ = Y.
(10) action (X, Y, cd):- door (X), not (closed (X)), interior (Y), X\ = Y.
(11) %Know if X is an existing place
(12) isPlace (X):- room (X).
(13) isPlace (X):- door (X).
(14) isPlace (X):- interior (X).
(15) isPlace (X):- cross (X).
(16) %Action rules than ensures reversibility
(17) isAction (X, X, Action):- action (X, X, Action).
(18) isAction (X, Y, Action):- action (X, Y, Action), X\ = Y, !.
(19) isAction (X, Y, Action):- action (Y, X, Action), X\ = Y, !.

```

SOURCECODE 2: Definition of action rules.

SLS and the MPS and the correspondent locations in the robot coordinate frame. When a task is triggered in a different location in which it is intended to be carried out, the robot guides the patient to the intended location. To do so, the robot localizes itself using the SLS. As a result, a semantic label is obtained. Then, the semantic label is looked up in the waypoints list. This way the robot is approximately localized. Next, the MPS is used to build a plan from the current location to the final destination. This plan is a list of waypoints, and the robot will try to reach one by one.

We relied in the *gmapping* ROS package to create the MPS waypoint list. This method reads the data provided by the laser and creates an occupancy grid map using it. The algorithms that comprehend *gmapping* are thoroughly explained in [38]. Despite being intended to be used by a laser sensor, we have not used the integrated laser sensors of

Pepper because of the lack of resolution, so we adopted the depth image to fake laser readings, as stated in [39].

The output of this step is a static 2D map that defines the limits where the robot can move through the walls, the doors, and the architectural barriers, but not the moving obstacles. In this map, we define the pose of the waypoints of the MPS, so we can translate the semantic locations to physical positions.

Once we have build the map of the environment, we can load and use it to perform the navigation. Then, we need to determine the position and the orientation of the robot within the map every time it moves. To do this, we used the Monte Carlo localization implemented in the adaptive Monte Carlo localization (AMCL) ROS package, which is explained in [40]. This method samples a set of particles in each iteration that represents a set of probable current poses

of the robot. It uses the information provided by the sensors to determine the validity of that prediction and concentrates the next predictions around the older most probably ones.

Finally, we have to resolve the path planning between the current pose of the robot and the target position and move the robot to the goal. This task is done using the ROS *move_base* package, one of the main elements in the ROS navigation stack. It receives a goal pose as input and then communicates with components such as the global and local planners, recovery behaviors, and costmaps and generates a velocity command for the base of the robot until it reaches the desired position. The components used by this node are explained in [41] and in the related links of that page (Figure 7).

Finally, it is worth noting that we used this mix of semantic localization and traditional mapping and navigation systems because the pure SLAM techniques tend to lose performance on the long term. It is very likely that the robot will not recover the localization once lost despite the accurate state-of-the-art SLAM methods. However, SLS provides an even more accurate localization method because it is based on visual features instead of laser features or odometry, which are often insufficient to provide a robust localization over time.

3.9. People Tracking and following Behavior. As aforementioned, the robot is intended to always stay by the patient in order to be noticed when it announces a scheduled task. To implement this behavior, we modeled it as a finite state machine as it is shown in Figure 8.

In the initial state, it starts by waiting for the patient to show in front of the robot and saying a customizable trigger statement. If the robot detects that statement using its speech recognition system explained in Section 3.5, then it tries to detect a person. To do so, we relied in YOLOv3-320 [42]. This is a region convolutional neural network architecture that is able to detect the position of the objects in the image plane, the label of those objects, and the correspondent detection score. This architecture achieved 0.51 mAP (measured over the intersection over union) over the test set of the COCO MS dataset. It is currently a state-of-the-art method on object detection and recognition providing a decent accuracy with low computation cost.

So, the robot uses its camera to forward the color data to this architecture in order to detect a person. If only one person is detected, the robot tries to maintain it in the center of its sight within a threshold by moving its base left or right. The robot also keeps a clear distance between itself and the patient of a customizable distance. This distance is computed using the front laser sensor of the robot. If the patient walks away, the robot must follow him by setting new goals to its navigation system, which is explained in Section 3.8, but always keeping the patient in the center of its sight and at the preset distance. It is worth noting that it will not move away when the patient approaches the robot. If the robot loses track of the person, it will be announced with a speech notification, and then, it will proceed to halt and try to detect a person one more time.

This behavior is kept until a programmed or contextual task is triggered, and navigation to a goal is required. In this case, the robot takes a role in which it is in charge of leading the patient to a destination room in order to perform the requested task. When the destination room is reached, the robot turns around (it assumes the patient is following it) and changes to the person detection role once again.

This simple yet effective tracking and the following system enables the robot to stay besides the patient at all times allowing a natural and fluent interaction.

4. Experimentation, Results, and Discussion

Before the deployment of our system in an actual scenario, the therapists and the persons in charge of the patient must define the tasks that the patient is able to perform. The tasks defined for this pilot experience are shown in Table 2. These tasks were suggested by therapists of ADACEA, which is a foundation for the acquired brain injured people.

Then, it is required to build the initial model of the SLS and the corresponding map for the MPS. Figure 9 shows the plan of the test house with their rooms. Note that the navigation system will not use the full map but the waypoints in order to set the next navigation goal. The SLS subsystem will provide a good approximate localization for the navigation step.

The BRE and ORE models were also trained beforehand. Our approach assumes that this step is already done and it can use the aforementioned maps and trained models.

In the following subsections, we provide experimentation of each piece that compose the system.

It is also worth noting that some of the experiments involved actual patients and their homes, but in some others, the patient had to be simulated by fellow research mates. This is due to lack of authorization from the patients.

Finally, as the computation power of Pepper robot's integrated processor is quite limited, the computation of the ORE, the BRE, and the execution of the YOLO architecture are performed in an additional computer equipped with a Nvidia GTX 1080Ti GPU. The robot and the additional computer are interconnected using the ROS framework. The neural architectures were developed using the Darknet and Keras frameworks.

4.1. Object Recognition Engine Experimentation. The architecture was trained ad-hoc for the ORE. In order to build the dataset, we downloaded the first 400 most relevant images with a public domain license from Google images for each object intended to be detected. These objects are those required by the tasks the patient must perform. So, the objects covered by the ORE are toothbrush, remote, bowl, toothpaste, bottle, egg, skillet, glass, and razor. The images were distributed in the training, validation, and test splits at 70%, 20%, and 10% each. The optimizer of choice was Adam, with a learning rate of 0.0001. The architecture was initialized with the ILSVRC 2012 model and trained for 10 epochs reaching a validation accuracy of 92.75% and a test accuracy of 92.61%. It is worth noting that the detection threshold was

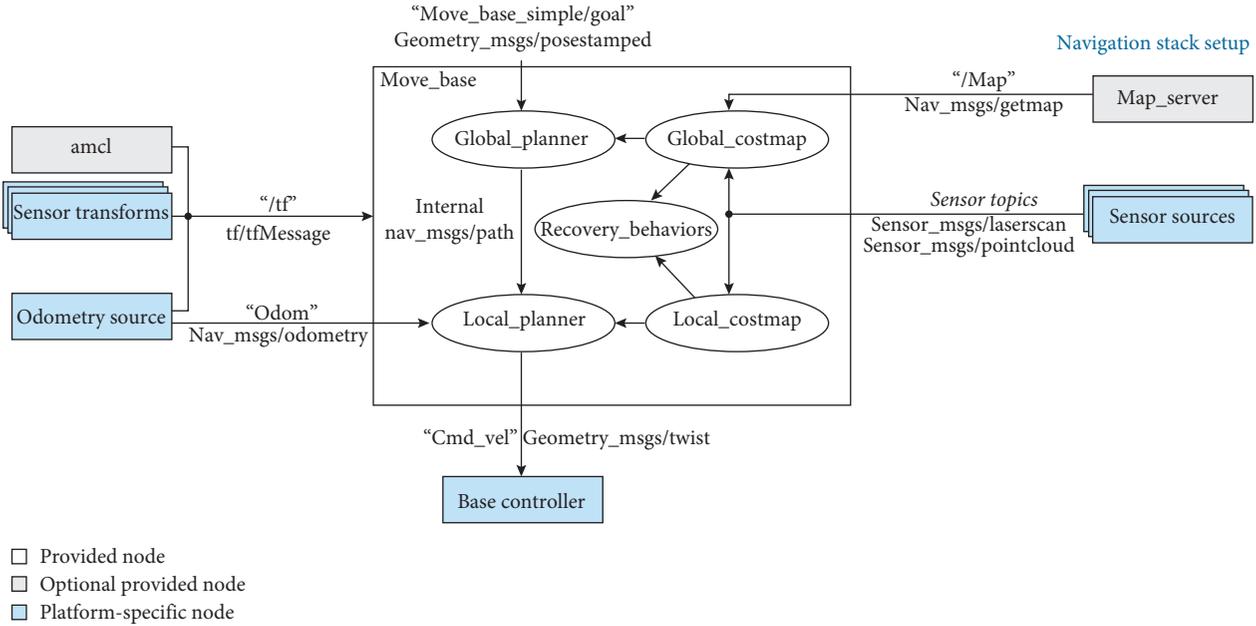


FIGURE 7: Global scheme of the navigation stack. Extracted from [41].

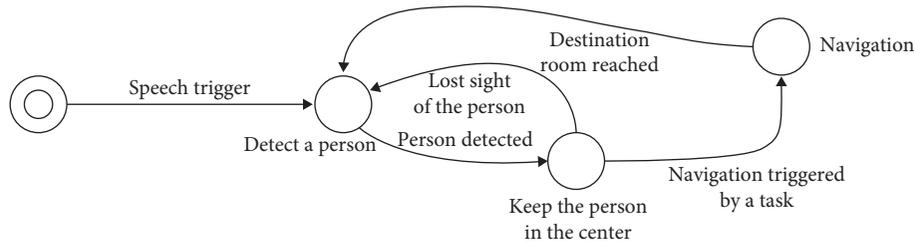


FIGURE 8: Finite states machine that models the people tracking and following behavior of the robot.

TABLE 2: Defined tasks for a test patient. Note that those tasks with no deadline in the scheduled column are contextual. The ORE goals define the objects that must be detected by the ORE. BRE goals define the behaviors that must be detected by the BRE. QRRE goals are the QR labels needed for that action.

Task	Location	Scheduled	ORE goals	BRE goals	QRRE goals
Shave	Bathroom	—	Razor and bottle	Shave	—
Drink a glass of water	Kitchen	—	Glass and bottle	Pour	—
Brush teeth	Bathroom	9:30	Toothbrush and toothpaste	Brush teeth	—
Water the plant of the bedroom	Bedroom	10:30	Plant and water	Pour	—
Cook scrambled eggs	Kitchen	12:30	Eggs, skillet, and bowl	Pour and beat	—
Turn the AC on	Living room	—	Remote	—	—
Workout: rise the arms	Bedroom	15:00	—	Rise arm	—
Take a painkiller	Bathroom	15:30	—	—	Painkiller

empirically set to 0.6. Figure 10 depicts the accuracy per class of the test split. Figure 11 depicts some samples of these objects correctly detected by the robot in the context of different task guidance.

4.2. Behavior Recognition Engine Experimentation. With the aim of evaluating the BRE’s performance, several subjects (nine in total; four women and five men) were recorded carrying out the five considered behaviors (i.e., shave, pour,

brush teeth, beat, and rise arms) in different scenarios. So, all the video sequences captured by the top Pepper’s RGB camera were divided into frames and manually labeled with the observed behavior. Then, these images were processed by Openpose [26, 27, 28]. After that, data augmentation was applied in order to properly identify left- and right-handed behaviors. The total of 25,286 images was used to train and test the classification ResNet50 [29] network. In particular, 75% of images were for the training and 25% of them were for the test. The optimizer choice was Adam, and the model

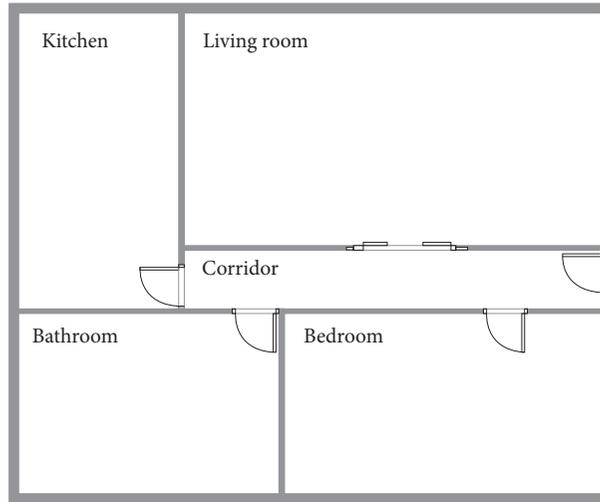


FIGURE 9: The actual plan of a test house.

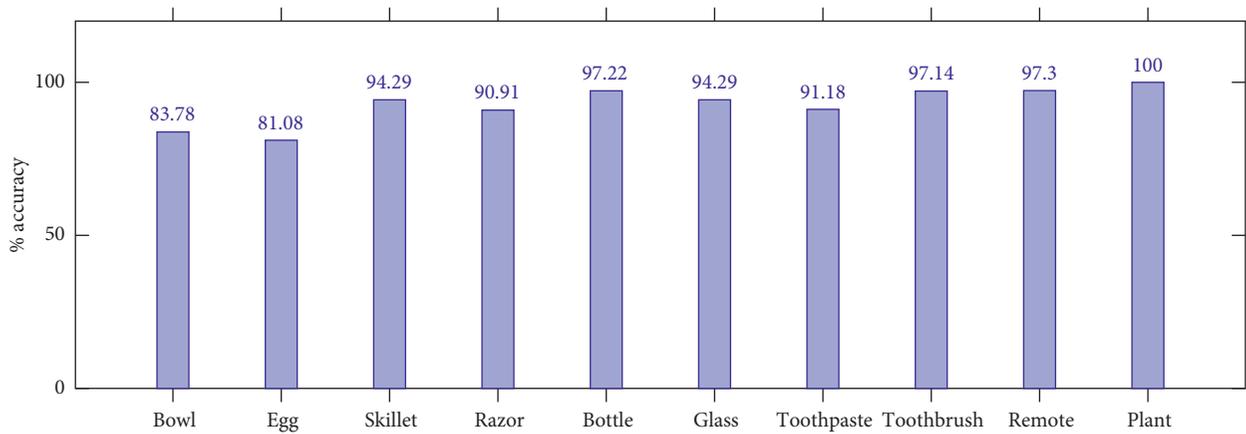


FIGURE 10: Accuracy per class distribution of the test split obtained by the object recognition engine.

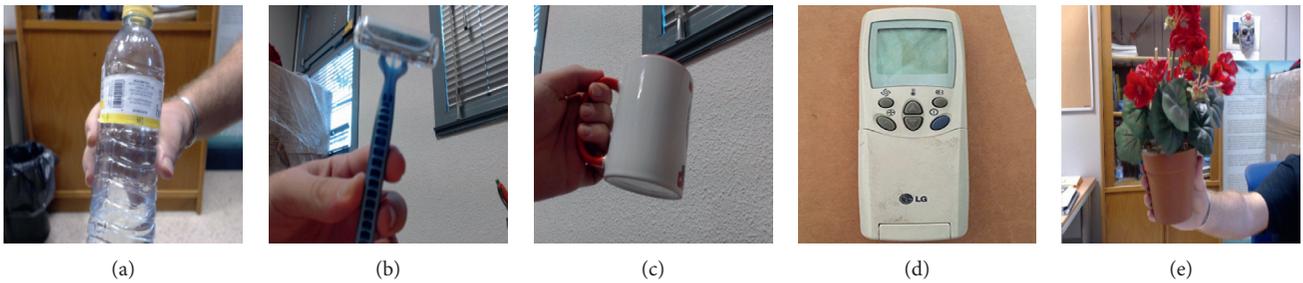


FIGURE 11: Samples of objects correctly detected by the object recognition engine. Note that these samples do not belong to any train, validation, or test split.

was trained for 150 epochs reaching a test accuracy of 99.98%. Figure 12 shows the confusion matrices obtained for the training and test.

4.3. *Semantic Localization System Experimentation.* As aforementioned, the objective of this module is to help the

robot get to know the location of places in houses. With this, the robot will be able to identify the place where it is.

In order to train the base model, we took video sequences from different residences and then randomly shuffled and distributed them into 70% training and 30% test splits. Table 3 shows the final number of samples per category. We use only RGB frames.

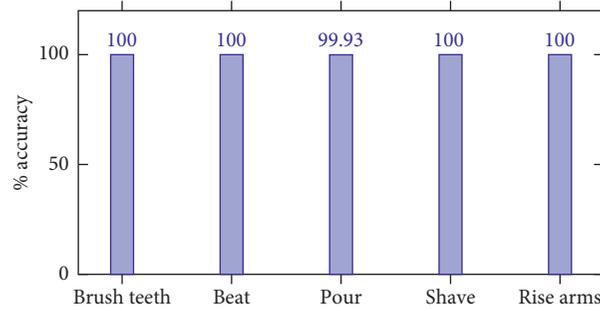


FIGURE 12: Accuracy corresponding to the test set of the behavior recognition engine after 150 epochs of training.

TABLE 3: Images distribution per category.

Cat. ID	Category	Training	Test
1	Corridor	4,747	2,037
2	Living room	6,205	2,661
3	Bathroom	3,299	1,415
4	Kitchen	4,878	1,616
5	Bedroom	6,205	3,216

The experimentation described in this section was carried out using our own dataset which provides a semantic category for each RGB image. It is important to state that the base model was built with images from four different residences. The categories come from the location in which the images were acquired. Figure 13 shows representative images for the 5 categories available in the dataset.

The experiments consist of measuring the performance of an already trained model with images from the house presented in Figure 9. It is worth noting that the cited model was trained on the dataset we described earlier.

For experiments in which new knowledge was included, we used images that were captured in the above-mentioned house. In this house, we have the same semantic categories but different visual appearances. The robot then proceeded to capture new information about the environment that the system had failed to identify. Subsequently, the new information was added to the current learned model.

First, we comment on the experiments carried out in the different rooms using only the base model, and then we discuss what happened when the system had flaws in the classification and we capture information from the new environment. A summary of the results for the experiments performed can be found in Figure 14.

Experiment 1 establishes the baseline we use to compare the following experiments. The total accuracy of the test is 94.91%. This represents the starting line, as no new knowledge was added.

Experiments from 2 to 6 were performed in above-mentioned house, obtaining results of (corridor \rightarrow 61.27%), (living \rightarrow 75.30%), (bathroom \rightarrow 36.61%), (kitchen \rightarrow 60.54%), and (bedroom \rightarrow 43.63%) when the robot did not know the environment and 100% in all the places once it had added information about these places.

The experimentation confirms the accuracy of the system and validates it for its deployment for semantic localization uses.

4.4. Motion Planning System Experimentation. For the experimentation of the motion planning system, we are using the concrete example of the house described in Figure 9, with the numerated nodes shown in Figure 15. We have defined the node types of every point in Sourcecode 3, so that every line number between 1 and 12 corresponds to the definition of the same numerated node. We have defined the connection between nodes in Sourcecode 4 with the associated cost and the direction that the robot must take to go from node A to B . As stated in Section 3.7, the definition of the connection between nodes is only made one-sided.

First of all, we calculate the paths from the kitchen to the others rooms, covering all the existing nodes in the graph. The results indicate that every node can be reached. To ensure the reliability of the rules that grant reverse connections, we calculate the reverse paths of the previous queries too.

Once we have checked that this system calculates all the paths and their reverses, we test the functionality of the dynamic predicate `closed/1`, so we cannot reach a goal following a path if there is some door closed.

We have covered all the possible paths between nodes in our experimentation. Due to space constraints, we only show as example the Execution Result 1.

4.5. Navigation and Mapping Experimentation. As stated in Section 3.8, we have used the *gmapping* algorithm in order to build the 2D static map of the environment, using a fake laser read from the depth sensor of Pepper. The results are shown in Figure 16. Additionally to the generated map, we have defined the position of the MPS nodes in order to



FIGURE 13: Sample images for each category of our home dataset: (a) corridor; (b) living room; (c) bathroom; (d) kitchen; (e) bedroom.

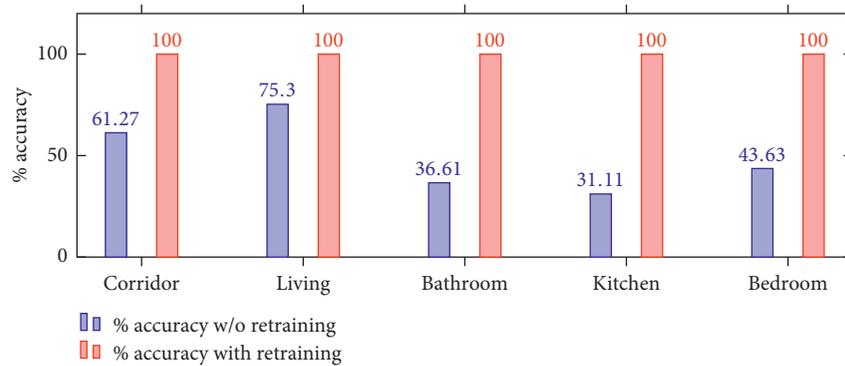


FIGURE 14: Accuracy of the SLS before and after applying the retraining process.

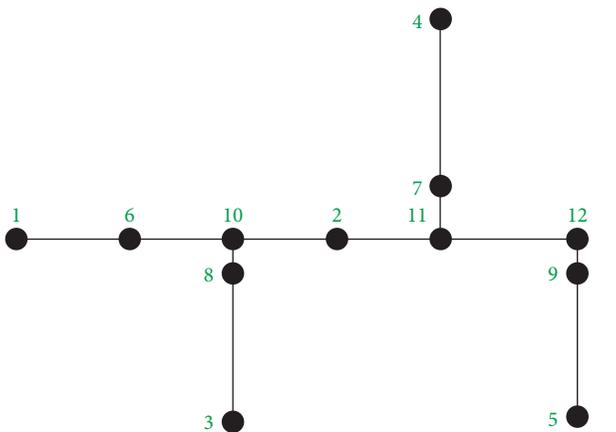


FIGURE 15: Numbered nodes of the house's graph representation.

associate a physical location to the semantic ones, so the robot can perform the navigation between the nodes.

Using this map, Pepper can perform the localization using the aforementioned *adaptive Monte Carlo localization* with its laser reads. As depicted in Figure 17, we can see the particles sampled by this algorithm with the most probable poses of the robot. When a well-identifiable location is captured by the laser, the density of the particles concentrates over its actual position.

The navigation has been successfully performed with the ROS *move_base* package. The costmaps generated by the planners according to the laser reads locate the dynamic obstacles and let them to compute the optimal path between the current pose and the goal. Additionally, Pepper incorporates an extra level of collision avoiding that blocks the movement of its base when the sonar detects an obstacle.

```

(1) %Type definitions
(2) room (kitchen).
(3) room (corridor).
(4) room (bathroom).
(5) room (living_room).
(6) room (bedroom).
(7) door (door_one).
(8) door (door_two).
(9) door (door_three).
(10) door (door_four).
(11) interior (interior_corridor_one).
(12) interior (interior_corridor_two).
(13) interior (interior_corridor_three).
(14) %Types without examples
(15) :-dynamic
(16) cross/1.
(17) %Dynamic predicate to indicate closed doors
(18) :-dynamic
(19) closed/1.
    
```

SOURCECODE 3: Definition of node types facts.

This security margin from the Pepper-integrated system makes door crossing difficult when the door is not quite big.

4.6. People Tracking and following Behavior Experimentation.

In this case, the architecture of choice was not trained from scratch, but we adopted an already trained model. This model was trained on the COCO MS dataset which is able to accurately detect persons among other objects. The detection of the rest of the objects is ignored, so we only retrieve the detection of the label person. This model is accurate enough

```

(1) % Connection between nodes
(2) connection (kitchen,door_one, east, 100).
(3) connection (door_one, interior_corridor_one, east, 100).
(4) connection (interior_corridor_one, door_three,south, 100).
(5) connection (door_three,bathroom, south, 100).
(6) connection (interior_corridor_one, corridor,east, 100).
(7) connection (corridor,interior_corridor_two, east, 100).
(8) connection (interior_corridor_two, door_two, north, 100).
(9) connection (door_two, living_room,north, 100).
(10) connection (interior_corridor_two, interior_corridor_three,east, 100).
(11) connection (interior_corridor_three,door_four, south, 100).
(12) connection (door_four,bedroom,south, 100).

```

SOURCECODE 4: Definition of node connections facts.

```

?- goToFrom (kitchen, bedroom, Path, Actions, Directions, Cost).
Path = [kitchen, door_one, interior_corridor_one, corridor, interior_corridor_two, interior_corridor_three, door_four, bedroom],
Actions = [cd, cd, fc, fc, fc, cd, cd],
Directions = [east, east, east, east, east, south, south],
Cost = 700.

```

EXECUTION RESULT 1: Execution result from kitchen to bedroom.

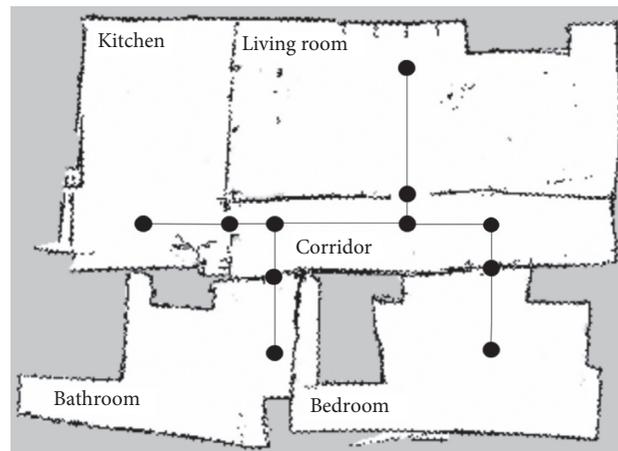


FIGURE 16: The image shows the actual map generated for the MPS with the correspondent graph superimposed. The room names correspond to the semantic labels used by the SLS. The map was iteratively generated by gmapping.

to detect persons in a variety of poses, even when the person is facing backwards the camera, sitting or lying in a bed or couch, or even if they are using a wheelchair as depicted in Figure 18. The accuracy of the architecture in these cases is specially important for our system because the patients are highly likely to often render these poses.

The tracking method tries to maintain the patient always in the center of the robot's sight within a threshold and a preset distance. In the experiments, the centering threshold was set to 70 px. This threshold enables a fine centering process while avoiding excessive movement of the robot due to little displacements of the patient or flickering in the detected area of the person. Figure 19 depicts the people tracking method. The clear space

between the patient and the robot was set to 70 cm, which places the robot far enough to enable the free movement of both robot and patient, while being closer enough to assure a fluent interaction. There is also a 10 cm threshold for the same reason we mentioned earlier. The speed of the linear movement is set to 0.3 m/s and the speed of self-rotation to 0.3 rad/s. We noticed that this method is highly dependent of the response time. In our test setup, the mean image acquisition time is 126 ms, whilst the person detection takes 301 ms mean. Both measures include the inter-communication overhead.

The people tracking and following behavior performed robustly. The robot only lost track of the patient when he moved unusually fast, so it completely fell out of sight of the

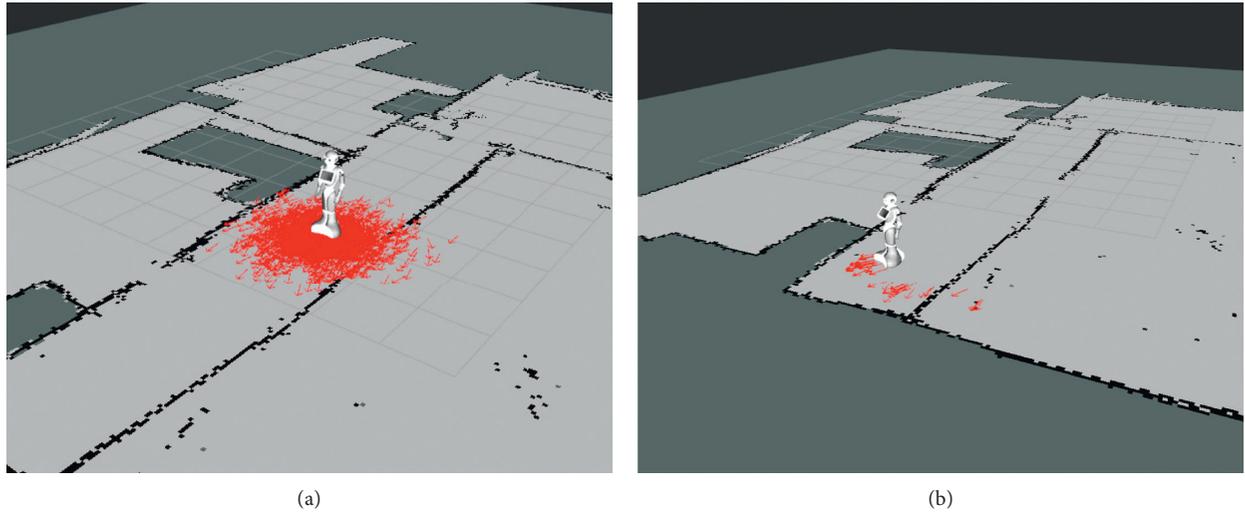


FIGURE 17: Adaptive Monte Carlo localization running. Every arrow represents a particle with an estimated 2D pose of the robot. (a) High uncertainty so there are multiple plausible poses of the robot (depicted as a big cloud of red arrows around the robot). (b) The robot saw a feature that helped to reduce the uncertainty, so the plausible poses are significantly reduced (shown as small clusters of red arrows around the robot).

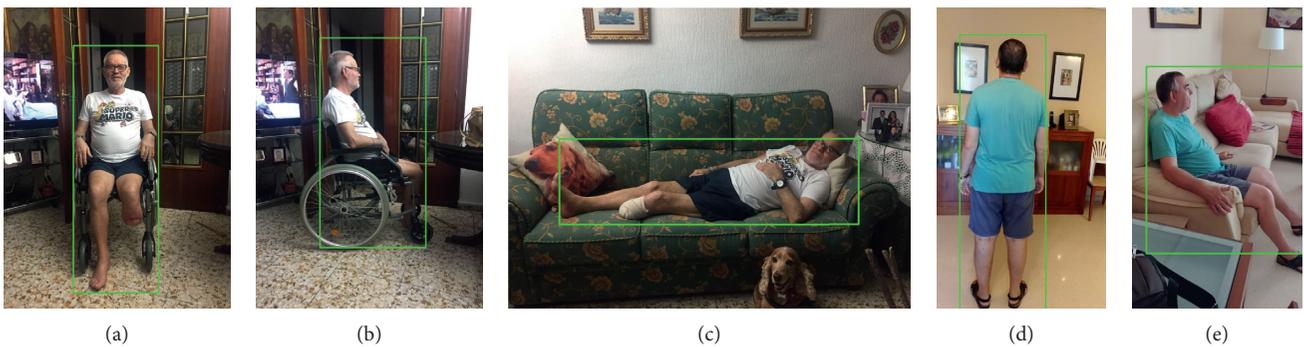


FIGURE 18: Some samples of the architecture performance, which is part of the people tracking subsystem. The robustness of the architecture is critical given the depicted cases, since the patients are highly likely to often render these poses.

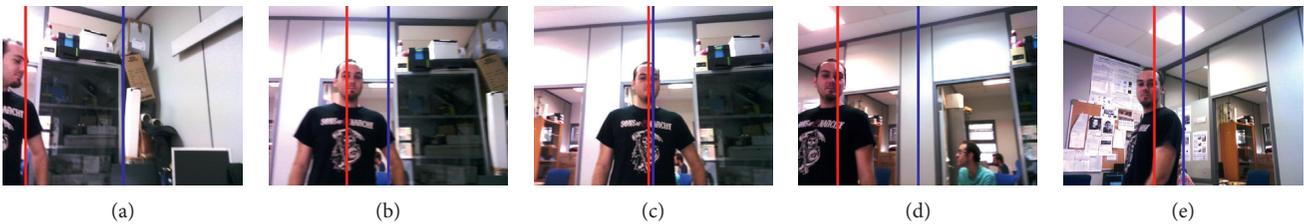


FIGURE 19: The robot tracks the person and always have him in the center of its sight by moving its base. The vertical blue line shows the center of the sight of the robot, whilst the red one depicts the center of the person. Both lines are aligned within a threshold.

robot. In this case, the robot asked the patient to position himself in front of it, and the tracking resumed properly.

5. Conclusions and Limitations

A robotic system for monitoring and interacting with people affected by cognitive diseases is proposed in this paper. The

system successfully integrates object recognition, activity recognition, localization, and navigation methods to remember and help the patients to perform their daily tasks.

Nonetheless, the system has some limitations. First, the initial stage where the map is created and the models are trained is mandatory and must be carried out by experts. In addition, the models for ORE and BRE have to be rebuilt if

new objects are required, as they are needed for so far unconsidered tasks that we would like to add to the patient's schedule. This issue could be mitigated by creating a proper plan that considers the long-term evolution of the patient on the first place.

The system was supervised by ADACEA, which is a foundation for the acquired brain injured people, that ensured it effectively may help the patients to improve their self-autonomy and quality of life.

Finally, it is worth noting that there is a video in the supplementary materials that depict the different subsystems running in test environments.

Data Availability

The image data used to support the findings of this study have not been made available because they involve vulnerable social groups, and the persons in charge of them refuse the access to the data without a formal request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been supported by the Spanish Government TIN2016-76515R Grant, supported with FEDER funds. Edmanuel Cruz is funded by a Panamanian grant for PhD studies IFARHU & SENACYT 270-2016-207. Jose Carlos Rangel was supported by the National System of Research (SNI) of the SENACYT of Panama. This work has also been supported by a Spanish grant for PhD studies ACIF/2017/243 and FPU16/00887. Thanks also to Nvidia for the generous donation of a Titan Xp and a Quadro P6000.

Supplementary Materials

Alongside the paper is attached a video demonstration of our proposal. In this video, the different pieces that integrate the system are depicted. First, the people tracking and following behavior is shown (Section 3.9). In this segment, a video from the point of view of the robot can be seen. The blue vertical line marks the center of the image, whilst the red one shows the center of the person, which is remarked in a green box. The characters in the left bottom corner represent the left, front, back, and right movement commands the system is issuing to the robot. Red is disabled, and green means the command is being sent. The user is wandering in a room, and the robot successfully tracks and follows the patient. The following segment shows the implementation of the programmed schedule and the context schedule. Both systems are explained in Section 3. The colors in the programmed schedule shows delayed or aborted tasks in red, imminent tasks in yellow, and the remaining tasks in blue. In this experiment, the localizations are being simulated by shuffling different locations. This is done to show the contextual schedule. Finally, two complete tasks are shown. First, a "take medicine A" task is due, so the robot asks the patient if he wants to attend the task. The user answers "OK," so the

robot uses its speech recognition capabilities (Section 3.5) to start the task. Then, the robot asks the patient to show the "medicine A." In this case, the QRRE is used (Section 3.4) to properly recognize the QR code of that medicine. The second task is "pour a glass of water." Once again the robot announces the task is due, and the patient triggers the task by answering "yes." The robot asks the user to show a glass and the water, which are being recognized properly by the ORE (Section 3.2). Then, the pour action is detected using the BRE (Section 3.3). The detection scores are superimposed in the top left corner. This video feed comes from the camera of the robot. Note that the robot detects that it is not the proper room to perform the task (kitchen), but the navigation (Section 3.8) is disabled as the testing environment does not feature a kitchen. (*Supplementary Materials*)

References

- [1] C. O'. Rooney, M. P. O. Hare Greg, R. D. Donoghue, and R. Brian, *What is a Social Robot*, University College Dublin, Dublin, Ireland, 1999.
- [2] S. Ikeda, F. Arai, T. Fukuda et al., "Vitro patient-tailored anatomical model of cerebral artery for evaluating medical robots and systems for intravascular neurosurgery," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1558–1563, Edmonton, AB, Canada, August 2005.
- [3] S. Ikeda, F. Arai, T. Fukuda, M. Negoro, K. Irie, and I. Takahashi, "Patient-specific neurovascular simulator for evaluating the performance of medical robots and instruments," in *Proceedings 2006 IEEE International Conference on Robotics and Automation*, pp. 625–630, ICRA, Orlando, FL, USA, May 2006.
- [4] C. Kouroupetroglou, D. Casey, M. Raciti et al., "Interacting with dementia: the mario approach," in *Studies in Health Technology and Informatics*, vol. 242, pp. 38–47, IOS Press, Netherlands, 2017.
- [5] W. Moyle, C. Jones, B. Sung et al., "What effect does an animal robot called cuddler have on the engagement and emotional response of older people with dementia? A pilot feasibility study," *International Journal of Social Robotics*, vol. 8, no. 1, pp. 145–156, 2016.
- [6] A. J. Rentschler, R. A. Cooper, B. Blasch, and M. L. Boninger, "Intelligent walkers for the elderly: performance and safety testing of va-pamaid robotic walker," *Journal of Rehabilitation Research and Development*, vol. 40, no. 5, pp. 423–431, 2003.
- [7] H. F. Machiel Van der Loos, J. Joseph Wagner, N. Smaby et al., "Provar assistive robot system architecture," in *Proceedings 1999 IEEE International Conference on Robotics and Automation*, pp. 741–746, Detroit, MI, USA, May 1999.
- [8] P. E. Hsu, Y. L. Hsu, K. W. Chang, and C. Geiser, "Mobility assistance design of the intelligent robotic wheelchair," *International Journal of Advanced Robotic Systems*, vol. 9, no. 6, p. 244, 2012.
- [9] M. Hillman, K. Hagan, S. Hagan, J. Jepson, and R. Orpwood, "The weston wheelchair mounted assistive robot—the design story," *Robotica*, vol. 20, no. 2, pp. 125–132, 2002.
- [10] E. Falcone, R. Gockley, E. Porter, and I. Nourbakhsh, "The personal rover project: the comprehensive design of a domestic personal robot," *Robotics and Autonomous Systems Socially Interactive Robots*, vol. 42, no. 3–4, pp. 245–258, 2003.
- [11] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun, "Towards robotic assistants in nursing homes: challenges and

- results,” *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 271–281, 2003.
- [12] M. E. Pollack, L. Brown, D. Colbry et al., “Autominder: an intelligent cognitive orthotic system for people with memory impairment,” *Robotics and Autonomous Systems*, vol. 44, no. 3-4, pp. 273–282, 2003.
- [13] M. Karlsson, P. Engelbrektsson, H. Hunter, A. M. O’Niell, H. Petrie, and D. Zoldan, “PAM-AID. personal adaptive mobility aid for the frail and elderly visually impaired. d3. 1. user requirement study,” Technical Report, Chalmers University, Gothenburg, Sweden, 1997.
- [14] J. Fasola and M. J. Mataric, “Robot exercise instructor: a socially assistive robot system to monitor and encourage physical exercise for the elderly,” in *Proceedings of 19th International Symposium in Robot and Human Interactive Communication*, pp. 416–421, Viareggio, Italy, September 2010.
- [15] S. Srinivasa, D. Ferguson, C. Helfrich et al., “Herb: a home exploring robotic butler,” *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.
- [16] T. Wang, H. Zhang, X. Ma, Y. Zhu, Z. Zhou, and B. Qian, “A home nursing robot system,” in *Future Intelligent Information Systems*, D. Zeng, Ed., Springer, Berlin, Heidelberg, 2011.
- [17] A. J. Huete, J. G. Victores, S. Martinez, A. Gimenez, and C. Balaguer, “Personal autonomy rehabilitation in home environments by a portable assistive robot,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 561–570, 2012.
- [18] S. S. Srinivasa, D. Berenson, M. Cakmak et al., “Herb 2.0: lessons learned from developing a mobile manipulator for the home,” *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2410–2428, 2012.
- [19] W. G. Louie, T. Vaquero, G. Nejat, and J. C. Beck, “An autonomous assistive robot for planning, scheduling and facilitating multi-user activities,” in *Proceeding of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5292–5298, Hong Kong, China, May 2014.
- [20] J. Ding, Y. Lim, M. Solano et al., “Giving patients a lift—the robotic nursing assistant (rona),” in *Proceedings of IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, pp. 1–5, Woburn, MA, USA, April 2014.
- [21] B. Görer, A. Ali Salah, and H. Levent Akin, “An autonomous robotic exercise tutor for elderly people,” *Autonomous Robots*, vol. 41, no. 3, pp. 657–678, 2017.
- [22] P. Lv, B. Zhou, and K. Wu, “A new remote health-care system based on moving robot intended for the elderly at home,” *Journal of Healthcare Engineering*, vol. 2018, Article ID 4949863, 11 pages, 2018.
- [23] A. Costa, E. Martinez-Martin, M. Cazorla, and V. Julian, “Pharos—physical assistant robot system,” *Sensors*, vol. 18, no. 8, p. 2633, 2018.
- [24] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, inception-ResNet and the impact of residual connections on learning,” 2016, <http://arxiv.org/abs/1602.07261>.
- [25] O. Russakovsky, D. Jia, H. Su et al., “ImageNet large scale visual recognition challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
- [26] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7291–7299, Honolulu, HI, USA, July 2017.
- [27] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, “Hand keypoint detection in single images using multiview bootstrapping,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4645–4653, Honolulu, HI, USA, July 2017..
- [28] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas Valley, NV, USA, June 2016.
- [29] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 2015.
- [30] J. Brown, *Zbar Bar Code Reader*, 2018, <http://zbar.sourceforge.net/>.
- [31] G. Saman and S. A. M. Gilani, “Object recognition by modified scale invariant feature transform,” in *Proceedings of Third International Workshop on Semantic Media Adaptation and Personalization*, pp. 33–39, Prague, Czech Republic, December 2008.
- [32] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [33] E. Rublee, V. Rabaud, K. Konolige, and G. B. Orb, “An efficient alternative to sift or surf,” in *Proceedings of International Conference on Computer Vision*, pp. 2564–2571, Tokyo, Japan, November 2011.
- [34] M. Rudinac, B. Lenseigne, P. Pieter, and Jonker, “Keypoint extraction and selection for object recognition,” in *Proceedings of Machine Vision Applications MVA*, Yokohama, Japan, May 2009.
- [35] Aldebaran, <http://doc.aldebaran.com/2-5/naoqi/audio/alspeechrecognition.html>, 2018.
- [36] E. Cruz, J. C. Rangel, F. Gomez-Donoso, Z. Bauer, M. Cazorla, and J. Garcia-Rodriguez, “Finding the place: how to train and use convolutional neural networks for a dynamically learning robot,” in *International Joint Conference on Neural Networks (IJCNN)*, pp. 3655–3662, Rio de Janeiro, Brazil, July 2018.
- [37] Annoy, <https://github.com/spotify/annoy>, 2018.
- [38] G. Grisetti, C. Stachniss, and W. Burgard, “Improved techniques for grid mapping with rao-blackwellized particle filters,” *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.
- [39] V. Perera, T. Pereira, J. Connell, M. Manuela, and M. M. Veloso, *Setting Up Pepper for Autonomous Navigation and Personalized Interaction with Users*, CoRR, [abs/1704.02767](https://arxiv.org/abs/1704.02767), 2017.
- [40] D. Fox, W. Burgard, D. Frank, and S. Thrun, “Monte carlo localization: efficient position estimation for mobile robots,” in *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI’99)*, Orlando, FL, USA, July 1999.
- [41] ROS, *Move_Base*, ROS, 2018, <http://wiki.ros.org/move.base>.
- [42] J. Redmon and A. Farhadi, “YOLOv3: an incremental improvement,” 2018, <http://arxiv.org/abs/1804.02767>.

Research Article

A Novel Artificial Organic Control System for Mobile Robot Navigation in Assisted Living Using Vision- and Neural-Based Strategies

Hiram Ponce , Ernesto Moya-Albor , and Jorge Brieva 

Universidad Panamericana, Facultad de Ingeniería, Augusto Rodin 498, Ciudad de México 03920, Mexico

Correspondence should be addressed to Jorge Brieva; jbrieva@up.edu.mx

Received 7 September 2018; Revised 31 October 2018; Accepted 7 November 2018; Published 2 December 2018

Guest Editor: Jose Garcia-Rodriguez

Copyright © 2018 Hiram Ponce et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Robots in assisted living (RAL) are an alternative to support families and professional caregivers with a wide range of possibilities to take care of elderly people. Navigation of mobile robots is a challenging problem due to the uncertainty and dynamics of environments found in the context of places for elderly. To accomplish this goal, the navigation system tries to replicate such a complicated process inspired on the perception and judgment of human beings. In this work, we propose a novel nature-inspired control system for mobile RAL navigation using an artificial organic controller enhanced with vision-based strategies such as Hermite optical flow (OF) and convolutional neural networks (CNNs). Particularly, the Hermite OF is employed for obstacle motion detection while CNNs are occupied for obstacle distance estimation. We train the CNN using OF visual features guided by ultrasonic sensor-based measures in a 3D scenario. Our application is oriented to avoid mobile and fixed obstacles using a monocular camera in a simulated environment. For the experiments, we use the robot simulator V-REP, which is an integrated development environment into a distributed control architecture. Security and smoothness metrics as well as quantitative evaluation are computed and analyzed. Results showed that the proposed method works successfully in simulation conditions.

1. Introduction

Nowadays, there is a dramatic increase in the aging of the population. It is expected that the number of people over 60 years will go from 962 million in 2017 to 1.4 billion in 2030 and 2.1 billion in 2050 [1]. Along with this increasing in elderly people and consequently higher life expectancy, there is a necessity of the creation of new care strategies. This problem is aggravated by the shortage of professional caregivers and their high costs. Some experts argue that it is desirable for elderly to stay in their own home with a certain level of independence and a sense of comfort and security. However, it requires to maintain an acceptable quality of life and independence capability [2].

Robots in assisted living (RAL) are an alternative to support families and professional caregivers with a wide range of possibilities to take care of elderly people. There are many issues where robots have high potential for assistance such as social isolation, diminishing independent living,

physical and cognitive impairment, loss of mobility, lack of recreation, and risk of falls. These problems can be tackled with different robot designs categorized as service, assistance, social, and rehabilitation robots [2].

Autonomous navigation is a challenging problem required in RAL due to the uncertainty and dynamics of the environments. A robot must be aware of where it is in relation to the surrounding environment and to localize itself during all the time. Hence, robot navigation skills must include different tasks as perception, exploration, mapping, localization, path planning, and path execution [3]. To accomplish this goal, the navigation system should replicate such complicated processes inspired on the perception and judgment of human beings. One approach to do so in robotic systems is the usage of vision sensors as fixed cameras located on the robot to process the video information for navigating in a given environment.

One of the challenges using vision sensors is the characterization of the 3D scene for computing features that can be

used in the navigation system. Different approaches have been proposed to solve this complex problem like monocular- and stereovision-based systems. For instance, in stereo-based systems, depth can be computed directly with the known leaks in limitation of detection, camera setup, and low speed of the system [4]. In contrast, monocular systems can exploit the geometric restrictions of the scene with the inherent generalization problem [5], and they can use the apparent motion of objects in scene computed, for example, with methods based on optical flow (OF), in order to estimate depth in the scene.

In monocular applications, OF approaches have shown advantages regarding other methods specially in the relationship between spatial and temporal gradient [6]. These OF family methods are widely used in robotic applications assuming that the apparent velocity of the brightness pattern varies smoothly almost everywhere in an image [7]. It has been used for estimating depth of the scene [8, 9], relative motion [10], and apparent velocity estimation [11, 12].

Moreover, artificial intelligence (AI) has been widely used for navigation of robots in assisted living, obtaining different levels of cognition: reasoning, decision making, and learning. Recently, convolutional neural networks (CNNs) are used in a wide variety of computer vision and robotic applications such as depth and distance estimation [13–16]. In addition, obstacle detection using OF is presented in [17], and the authors in [18] developed distance estimation from robot to obstacles using CNN.

To get benefit from the combination of artificial intelligence and visual perception methods in RAL, this paper presents a novel nature-inspired control system for mobile robot navigation using an artificial organic controller enhanced with vision and neural-based strategies, i.e., Hermite OF and CNN using a monocular camera and tested in a simulated environment. Particularly, the Hermite OF is employed for obstacle motion detection while CNN is occupied for obstacle distance estimation. For the experiments, we use the robot simulator V-REP [19], which is an integrated development environment into a distributed control architecture. The simulated robot is equipped with only a monocular camera.

The contribution of this work is focused on the integration of raw OF features and the estimated object distance to the obstacle as input to the controller. The estimation of the object distance is performed using CNN. The latter is trained using the OF features as input and as reference for the distance measured using ultrasonic sensors. In addition, the whole set of features is used as input to the nature-inspired control system based on the artificial organic controller. We developed our approach from previous studies: a system employing a basic controller using only OF features [20] and the same optical features as input to an artificial organic controller [21]. All the experiments have been carried out using the same camera parameters.

The rest of the paper is arranged as follows: Section 2 describes the proposed approach, the experiments, and the protocol, and Section 3 discusses results and finally, conclusion is presented.

2. Materials and Methods

In this section, we present relevant studies associated to monocular depth estimation found in the literature. Then, it is explained the methods used in our approach and finally, the nature-inspired control system for mobile robot navigation is proposed.

2.1. Related Work. Monocular depth estimation can be tackled by CNN in two different ways: supervised and unsupervised. For instance, unsupervised methods include several approaches as discussed in [22], and the authors proposed a monocular depth estimation using a CNN trained without ground truth data. This proposal considered to exploit epipolar geometry constraints, giving as result a better depth map than traditional supervised learning methods. Another approach based on CNN and random forest is the proposal explained in [23]. The work presented a network trained by learning the parameters in an unsupervised way through maximizing the likelihood of the training data. In [24], a method using deep CNN to depth prediction without requiring a pretraining stage was reported. In [25], the authors proposed an unsupervised CNN-based method for explicit depth estimation from light field, which learns an end-to-end mapping from a 4D light field to the corresponding disparity map without the supervision of ground truth depth.

Several applications using supervised learning are presented in the literature. In [26], the authors described a depth estimation from monocular images using regression on deep features from a CNN and a conditional random field. The implementation considered two levels of depth inferring, pixel-to-pixel and regions of pixels. A similar approach was done in [27] but with discrete mapping inference. Gan et al. in [28] performed an explicit model to describe the relationships of images obtained from a monocular camera with an affinity layer and by combining absolute and relative features into a CNN, also local vertical features of depth estimation were incorporated. Cheng et al. [29] proposed a convolutional spatial propagation network (CSPN) to learn the affinity matrix for depth estimation from a single image. In [30], it is used a deep model to generate dense depth maps from a RGB image employing depth estimation of sparse set of pixels. In [31], deep structured model was presented in which the structured pixelwise depth estimation has ordinal constraints introduced by the user. In [32], the problem of estimating the depth map of a scene given in a single RGB image was solved by training a convolutional residual network to model the ambiguous mapping between monocular images and depth maps. However, most of the studies based on CNN suppose rigid scenes, as in the proposed method from [33]. In [34], depth map prediction employed two deep network stacks: the first makes a coarse global prediction on the whole image and in the second step the prediction is refined locally on the image.

Typically, monocular depth estimation considers that the captured scene is static and with constant depth. In practice,

there is a blurring effect between camera and the objects. Thus, other methods such as deblurring and flow estimation are required, as shown in [35]. However, the approach reported the usage of stereo cameras. Other approaches for dynamic scenes are those based on motion estimation. In this case, OF between two consecutive images is also applied for depth estimation through motion segmentation, as proposed in [36]. In [37], a method to automatically estimate the depth of video frames of a single camera was proposed, and this estimation was carried out by analyzing the OF of preexisting videos and by using a pretrained CNN.

An example of robot navigation and localization using monocular depth estimation can be found in [38]. It showed that using CNN for depth estimation combining with monocular simultaneous localization and mapping (SLAM) can be successfully applied.

2.2. Optical Flow. OF is a 2D distribution of apparent velocities associated, usually, with intensity pattern variations in a sequence of images, and it is represented by a vector field that encodes the displacement for each pixel in the image sequence.

There are many approaches to obtain a dense and accurate OF estimation, where it is well known that the differential methods overcome other ones [39]. Those are based on the work of Horn and Schunck [7], which assume that the intensity of the objects remains constant during small periods of time and that the neighboring pixels have similar displacement. This method has low computational time, but it cannot handle large displacements. Recent OF approaches are more accurate to large displacements, but they are computationally expensive and very difficult to implement in devices with limited hardware.

In this paper, OF proposal is based on the studies of Moya-Albor et al. [20] and Ponce et al. [21]. It uses the Hermite transform [40], as bioinspired image model, to incorporate local descriptors allowing to describe the intensity and gradient constraints found in the current methods. This model increases the accuracy of the Horn and Shunck method, and it is more robust to noise.

2.3. Convolutional Neural Networks. CNN is a well-known neural network architecture inspired on the nature of visual perception in living creatures [41] typically applied for classification and regression in image processing [42]. There exists different architectures of CNN, but it is mainly constituted by three types of layers, namely, convolutional, pooling, and fully connected. The first layer aims to compute feature representations of the input, a pooling layer aims to reduce the resolution of feature maps, and a fully connected layer aims to perform high-level reasoning [41]. Lastly, a CNN may include an output layer aiming to compute the classification or regression task. Particularly, image and video applications have been widely explored with CNN.

2.4. Artificial Organic Controllers. An artificial organic controller (AOC) is an intelligent control strategy aiming to

compute the control law using an ensemble method, namely, fuzzy-molecular inference (FMI) system [43]. It consists of a hybrid method from both fuzzy logic and artificial hydrocarbon networks (AHN). To properly design the AOC for the proposed robot system, an overview of AHN as well as the FMI system is introduced as follows.

2.4.1. Overview of Artificial Hydrocarbon Networks. In machine learning, AHN algorithm is a supervised learning method inspired on the inner mechanisms and interactions of chemical hydrocarbon compounds [44]. This method aims to model data points like packages of information, called molecules. The interaction among these units allows capturing the nonlinearities of data correlation. From this point of view, an artificial hydrocarbon compound is built, and it can be seen as a net of molecules. If required, more than one artificial hydrocarbon compound can be added up to finally get a mixture of compounds [45].

In AHN, the basic unit of processing information is the molecule. It performs an output response $\varphi(x)$ due to an input $x \in \mathbb{R}^k$, as expressed in Equation (1) where $v_C \in \mathbb{R}$ represents a carbon value, $h_{i,r} \in \mathbb{C}$ are the hydrogen values attached to this carbon atom, and d represents the number of hydrogen atoms in the molecule.

$$\varphi(x) = v_C \sum_{r=1}^k \prod_{i=1}^{d \leq 4} (x_r - h_{i,r}). \quad (1)$$

If two or more molecules have less number of hydrogen than allowed, i.e., $d < 4$, then they are able to join together forming chains of molecules. These chains are namely hydrocarbon compounds. Throughout this work, compounds are made of n molecules: a linear chain of $(n-2)\text{CH}_2$ molecules with two CH_3 molecules, one at each side of the CH_2 -chain [45]. In addition, a piecewise function ψ denoted as Equation (2) is associated to the compound representing its behavior due to an input x , where $L_t = \{L_{t,1}, \dots, L_{t,k}\}$ for all $t = 0, \dots, n$ are bounds where molecules can act over the input space [45].

$$\psi(x) = \begin{cases} \varphi_1(x), & L_{0,r} \leq x_r < L_{1,r}, \\ \dots & \dots \\ \varphi_n(x), & L_{n-1,r} \leq x_r \leq L_{n,r}. \end{cases} \quad (2)$$

Lastly, different compounds can be selected and added up to form complex structures called mixtures. In AHN, a mixture is a linear combination of behavior compounds ψ_j in finite ratios α_j , representing the weights of compounds, as expressed in the following equation:

$$S(x) = \sum_{j=1}^c \alpha_j \psi_j(x). \quad (3)$$

To this end, AHN is trained using the so-called AHN algorithm reported with detail in the literature [44–47].

2.4.2. Fuzzy-Molecular Inference System. As mentioned above, FMI is an ensemble of fuzzy logic and AHN [43].

Figure 1 shows the block diagram of the FMI. It consists of three main steps: fuzzification, fuzzy inference engine, and defuzzification based on AHN.

Fuzzification and fuzzy inference engine steps are quite similar to fuzzy logic. An input x is mapped to a set of fuzzy sets, using membership functions. Then, an inference operation, represented as a fuzzy rule, is applied to obtain a consequent value y_p . Considering the p th fuzzy rule R_p denoted as Equation (4), inference computes y_p in terms of an artificial hydrocarbon compound with n molecules, M_j , each one with function compound $\varphi_j(x)$ for all $j = 1, \dots, n$. In this work, the membership value of y_p is calculated using the min function, expressed as $\mu_{\Delta}(x_1, \dots, x_k)$, over the fuzzy inputs.

$$R_p: \text{if } x_1 \in A_1 \wedge \dots \wedge x_k \in A_k, \quad (4)$$

$$\text{then } y_p = \varphi_j(\mu_{\Delta}(x_1, \dots, x_k)).$$

In the defuzzification step, it computes the crisp output value y , using the center of gravity approach [43], as expressed in the following equation:

$$y = \frac{\sum_p \mu_{\Delta}(x_1, \dots, x_k) \cdot y_p}{\sum_p \mu_{\Delta}(x_1, \dots, x_k)}. \quad (5)$$

2.5. Nature-Inspired Control System for Mobile Robot Navigation. In this work, we propose a nature-inspired controller system for mobile robot navigation implementing an AOC enhanced with Hermite OF and CNN. Particularly, this approach requires only a single camera mounted in the robot, implementing in this way an ego-centric vision system. No other sensors are required for this controller. Figure 2 shows the block diagram of the proposed control system. It consists of the following steps: (i) motion object detection, (ii) distance object estimation, and (iii) control law computation.

2.5.1. Motion Object Detection. The motion object detection step considers determining the relative displacement that an object located in front of the robot is performing. This information is very useful when dealing with mobile obstacles. To compute the estimated relative displacement of the object in an image, the Hermite OF method is employed in a similar way as proposed in [21].

First, two adjacent gray images, I_t in time t and I_{t+k} in time $t+k$, are acquired by the single camera. Then, the Hermite OF algorithm computes the relative displacements of objects between these images. This procedure outputs a map of displacements. These relative displacements are decomposed in both horizontal (u) and vertical (v) components. Assuming that the mobile obstacle presents more displacement than the rest of the scene, then a mean value per axis, \bar{u} and \bar{v} , is calculated for estimating the relative displacement of the object. To this end, the mean values \bar{u} and \bar{v} are passing as inputs to the AOC explained below.

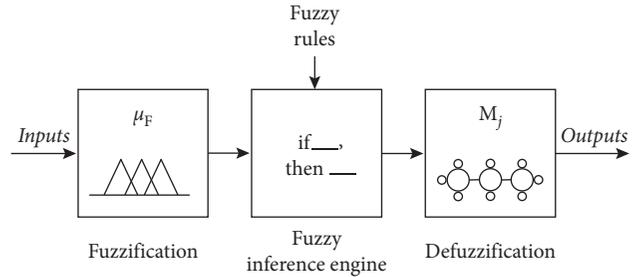


FIGURE 1: Block diagram of the fuzzy-molecular inference system.

2.5.2. Distance Object Estimation. The distance object estimation step considers determining the distance of the robot from an object using the single camera. However, computing this value in an image is a challenging task [4]. In this way, we propose to use a CNN for estimating the distance of an object in the environment.

To do so, the same two images I_t and I_{t+k} are occupied in this step. Once again, the Hermite OF method is computed using these images. The output map of displacements obtained with this method is converted to two images I_u and I_v , related to the horizontal and vertical displacements, respectively. Considering that the size of the original images I_t and I_{t+k} is $m \times n$, we use I_u and I_v to form a new image $I_{uv} = [I_u, I_v]$ of size $m \times 2n$. Then, the latter image is used as the input of the CNN.

The CNN has a convolutional layer with f filters of size $h \times h$ that is used for calculating the feature representations of the input image. It is followed by a rectified linear unit (ReLU) layer, and finally it has a fully connected layer, with output size 1, together with a regression layer to perform a high-level reasoning for estimating the distance \tilde{d} to an object. To implement the CNN, we previously obtained the features of motion leaving to the CNN only the reasoning process from the spatial localization of motion to the distance estimation. In this case, the proposed architecture could be minimal, and the size of the training data could also be small enough to learn robust features. Figure 3 shows the topology of the proposed CNN in the distance object estimation step. For training purposes, we prepared a dataset using ultrasonic sensors as target values, as explained later on. To this end, the distance estimation \tilde{d} is passed as input to the AOC explained below. No other architectures were tried for this work.

2.5.3. Control Law Computation. The last step of the nature-inspired control system is the control law computation using the AOC. As shown in Figure 2, three inputs are defined as follows: the mean values \bar{u} and \bar{v} , representing the relative displacement components of a mobile object, are partitioned in three fuzzy sets like “negative” (N), “zero” (Z), and “positive” (P), while the distance estimation \tilde{d} , from the robot to an object, is partitioned also in three fuzzy sets like “small” (S), “medium” (M), and “large” (L). Particularly for this work, the proposed input membership functions are depicted in Figure 4.

In addition, Table 1 presents the set of fuzzy rules designed for the mobile robot navigation task. These rules consider obstacle avoidance and free navigation of the robot. To this end, Figure 5 shows the artificial hydrocarbon compound developed for this work. It comprises three

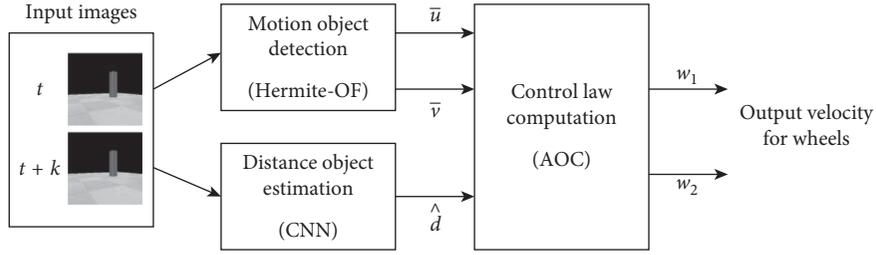


FIGURE 2: Block diagram of the nature-inspired control system.

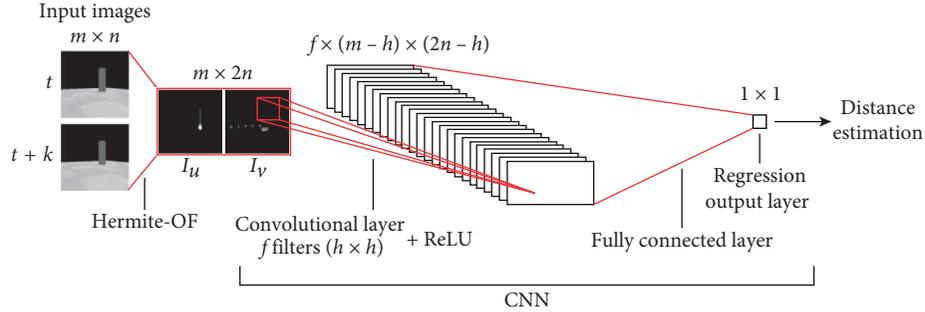


FIGURE 3: CNN topology of the proposed distance object estimation approach.

molecules representing the output velocity of the wheels w_1 and w_2 in the robot like “counterclockwise” (CCW), “stop” (S), and “clockwise” (CW).

3. Results and Discussion

In order to validate our proposed nature-inspired controller system for mobile robot navigation implementing an AOC enhanced with Hermite OF and CNN, we develop a set of experiments to independently prove each of the components of the system in an incremental fashion. These experiments measure the output response of (i) avoiding a mobile obstacle using the Hermite OF, (ii) avoiding a mobile obstacle as well as free navigating using the Hermite OF and AOC, (iii) avoiding a fixed obstacle using CNN, and (iv) avoiding fixed and mobile obstacles as well as free navigating using the whole proposed nature-inspired controller system.

In this work, the performance of the robot navigation is evaluated objectively by computing some metrics related to the security and smoothness of the control navigation response. Three security indexes are used to evaluate the distance between the robot trajectory and the location of obstacles [48]:

- (i) SM1: it measures the mean distance between the trajectory of the robot to the closest obstacle.
- (ii) SM2: it measures the minimum distance between the trajectory of the robot and the mean distance to all obstacles.
- (iii) SM3: it measures the minimum distance over the trajectory of the robot to the closest obstacle.

For the security metrics, larger values of the indexes represent a better behavior in the robot navigation, since

they intuitively measure the security distance at which the robot is located away from the obstacles.

In addition, three smoothness indexes are employed to indirectly evaluate the consistency between the decision-action relationship of the control navigation in the robot and the ability to react to events with sufficient speed [48]. The bending energy (B_E) measures the energy for steering or bending during the trajectory, and it is calculated as Equation (6), where k_t represents the curvature of the trajectory $f(t)$ computed as Equation (7), n is the number of points in the discrete trajectory, and t is the current time.

$$B_E = \frac{1}{n} \sum_{t=1}^n k_t^2, \quad (6)$$

$$k_t = \frac{f''(t)}{(1 + f'(t)^2)^{3/2}}. \quad (7)$$

The smoothness metric that considers bending energy over time (TB_E) is calculated as follows:

$$TB_E = \sum_{t=1}^n k_t^2. \quad (8)$$

Lastly, the smoothness of curvature (S_k) measures the change in curvature k all along the trajectory f with length L performed by the robot navigation over time, and it can be expressed as follows:

$$S_k = \frac{\int_0^L (dk/dt)^2 ds}{t}. \quad (9)$$

For smoothness metrics, smaller values close to zero represent smooth curvatures in trajectory and less energy in the performance.

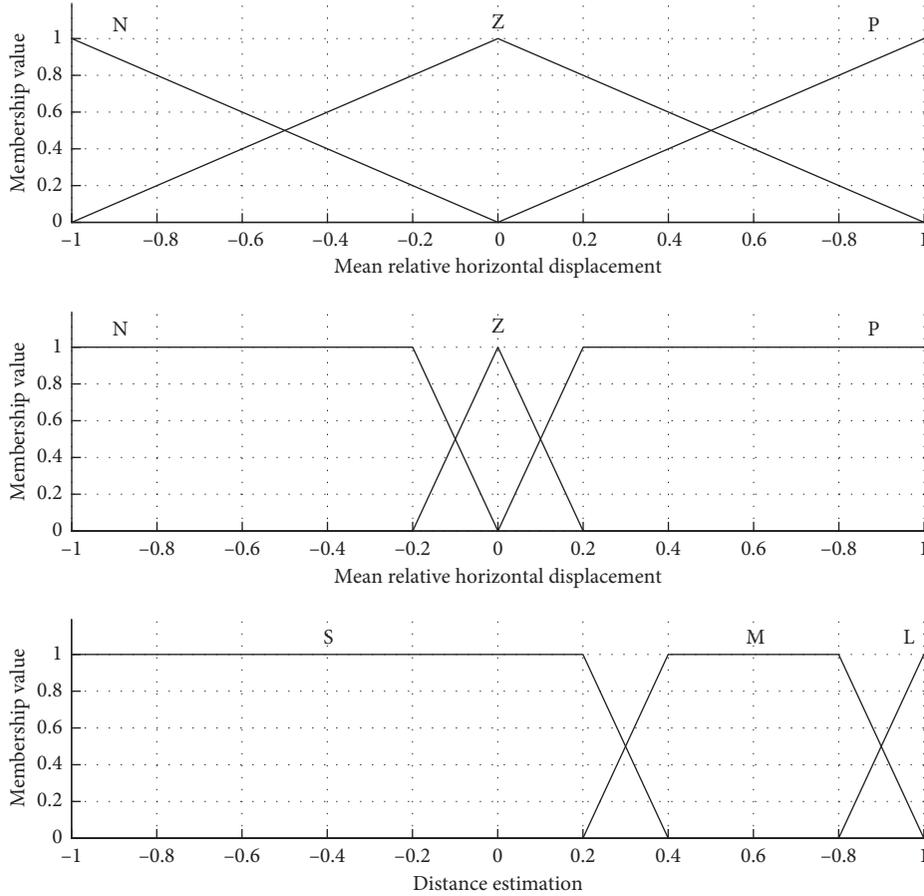


FIGURE 4: Input membership functions in the AOC: (a) mean relative horizontal displacement, (b) mean relative vertical displacement, and (c) distance estimation.

3.1. Mobile Obstacle Avoidance Using Real-Time Hermite OF.

The first experiment aims to measure the output response of avoiding mobile obstacles using the real-time Hermite OF. Figure 6 shows the initial configuration of the environment. It considers two mobile robots as obstacles (green and blue). The red robot shown in the scene is the one with the RT-HOF. For implementation purposes, this method is based on the work of [20].

In a nutshell, two consecutive images are acquired, and the Hermite OF is computed getting the relative displacements decomposed in horizontal u and vertical v components. Then, the mean values of the components, \bar{u} and \bar{v} , are calculated. In addition, the relative direction θ between vectors u and v are computed such that the mean angle $\bar{\theta}$ is obtained. Lastly, a simple set of rules are considered for avoiding mobile obstacles, as shown in Algorithm 1. In this work, the threshold values in the set of rules were set experimentally:

$$T_u = 0.05, T_v = -0.2, T_l = 40, T_u = 130, T_{m_1} = 90, \quad \text{and} \quad T_{m_2} = 130.$$

In Figure 7, it is shown the output trajectories when using the RT-HOF method. Five attempts were run (reported as the dashed red lines), and the mean trajectory (strong-red line) is depicted in Figure 7(a), while Figure 7(b) shows the speed of the target robot over its trajectory. Notice

that the red robot steers to the left trying to avoid the blue robot that is going to the right. In this particular case, the avoidance procedure controller (Algorithm 1) was designed for avoiding obstacles in the backwards. In addition, the trajectory of the target robot when dealing with the green robot is done by positioning in parallel to the direction of it. From Figure 7(b), it is observed that the red robot decreases its velocity once it detects another mobile object. Also, the velocity is discrete since the controller is based on a set of crisp rules. Moreover, Figure 8 shows the inputs (\bar{u} , \bar{v} , $\bar{\theta}$) and output (*speed*) values of the controller, where speed is related to the linear velocity of the target robot. It can be observed that \bar{u} has more influence than \bar{v} when objects are close to the robot. In addition, $\bar{\theta}$ is correlated to the steering action of the robot.

In addition, security and smoothness indexes are summarized in Table 2 for each of the attempts and the mean performance. As shown in the indexes, security values consider that the trajectories obtained allow the robot to be far from the mobile obstacles (mean minimum distance reached of 0.63 m). Smoothness indexes are close to zero meaning that the trajectories of the robot do not represent abrupt changes.

To this end, in this experiment, the target robot did not collide with any of the mobile obstacles in all the attempts.

TABLE 1: Fuzzy rules of the AOC for mobile robot navigation.

\bar{u}	\bar{v}	\hat{d}	w_1	w_2
N	N	S	M_{CW}	M_{CCW}
N	N	M	M_{CW}	M_S
N	N	L	M_{CW}	M_S
N	Z	S	M_{CCW}	M_{CW}
N	Z	M	M_S	M_{CW}
N	Z	L	M_{CW}	M_{CW}
N	P	S	M_{CCW}	M_{CW}
N	P	M	M_S	M_{CW}
N	P	L	M_S	M_{CW}
Z	N	S	M_{CW}	M_{CCW}
Z	N	M	M_{CW}	M_S
Z	N	L	M_{CW}	M_S
Z	Z	S	M_{CCW}	M_{CW}
Z	Z	M	M_S	M_{CW}
Z	Z	L	M_{CW}	M_{CW}
Z	P	S	M_{CCW}	M_{CW}
Z	P	M	M_S	M_{CW}
Z	P	L	M_S	M_{CW}
P	N	S	M_{CW}	M_{CCW}
P	N	M	M_{CW}	M_S
P	N	L	M_{CW}	M_S
P	Z	S	M_{CCW}	M_{CW}
P	Z	M	M_S	M_{CW}
P	Z	L	M_{CW}	M_{CW}
P	P	S	M_{CCW}	M_{CW}
P	P	M	M_S	M_{CW}
P	P	L	M_S	M_{CW}

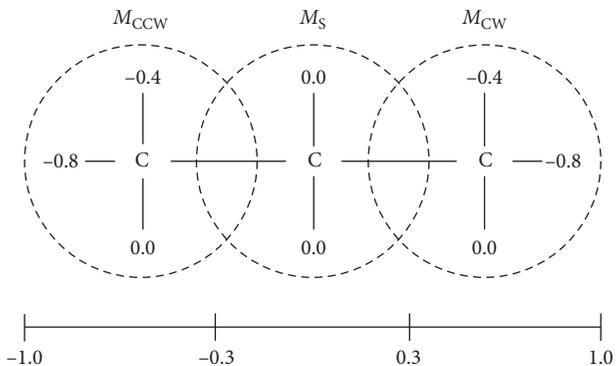
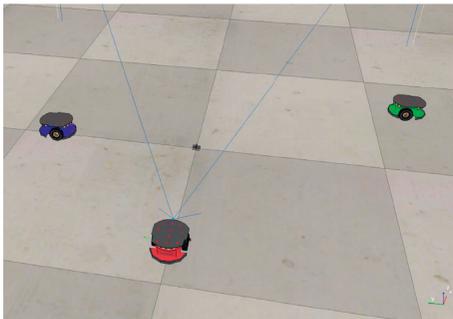

 FIGURE 5: Output molecular partitions in the AOC for the velocity calculation in both wheels w_1 and w_2 .


FIGURE 6: Scenario developed for testing avoidance of mobile obstacles (green and blue robots).

3.2. *Mobile Obstacle Avoidance and Free Navigation Using AOC.* Using the AOC with real-time Hermite OF and the inferred distance, from the CNN training step, as inputs, we tested the wheeled robot such that it can navigate freely in an environment avoiding obstacles.

Figure 9 shows the initial configuration of the environment. It considers one mobile robot as obstacle (blue). The red robot shown in the scene is the one with the RT-HOF method and the AOC. For implementation purposes, this method is based on the work of [21].

The output trajectories when using RT-HOF and AOC are shown in Figure 10. Five attempts were run (reported as the dashed red lines), and the mean trajectory (strong red line) is also depicted in Figure 10(a). Again, Figure 10(b) shows the velocity of the target robot over its trajectory. In this experiment, the target robot outputs a trajectory that prevents collision with the mobile obstacle (running from left-to-right) by reducing the velocity and steering a little bit to the left. Then, it corrects its trajectory, but different decisions are taken. It mainly happens due to the slight variations in the position of the mobile obstacle. Moreover, the velocity appreciated in Figure 10(b) is smoother than in the previous experiment. It is important for robotics implementations because it better regulates electrical current in the actuators. Figure 11 shows the inputs (\bar{u} , \bar{v} , $\bar{\theta}$) and output (*speed*) values for this controller.

In terms of the security and smoothness indexes, Table 3 summarizes the results for each of the attempts and the mean performance. As shown in the indexes, security values consider that the trajectories obtained allow the robot to be far from the mobile obstacles (mean minimum distance reached of 0.78 m). Smoothness indexes prove minimal abrupt changes in the trajectories.

Again, in this experiment, the target robot did not collide with the mobile obstacle in all the attempts.

3.3. *Fixed Obstacle Avoidance Using CNN.* This experiment includes the proposed CNN with OF distance-object estimation model for an egocentric vision-based robot. The CNN was trained from scratch using the data ($\mathcal{D}_{\text{cnn+of}}$) collected from a set of 10 attempts, with 860 samples each, between the robot and one fixed object. The dataset is balanced in terms of short and large distances. The dataset was divided into 70% training and 30% testing, randomly chosen. Then, images I_u and I_v representing the horizontal and vertical components of the OF were resized to 28×28 , and then concatenating them to produce a single 28×56 image, $I_{uv} = [I_u, I_v]$.

To this end, the CNN with OF was trained by using the pairs $\{I_{uv}, d\}_i$ in the training set, where d is the sensor-based distance measure, at the i -element of the set. Lastly, the stochastic gradient descent method was used for training with initial learning rate 0.01, momentum 0.90, and mini-batch size of 16. L_2 regularization was performed with $\lambda = 0.0001$ term.

After that, the root mean square error (RMSE) from Equation (10) was employed for measuring the performance of the CNN over the remaining 30% of the dataset, where d_q

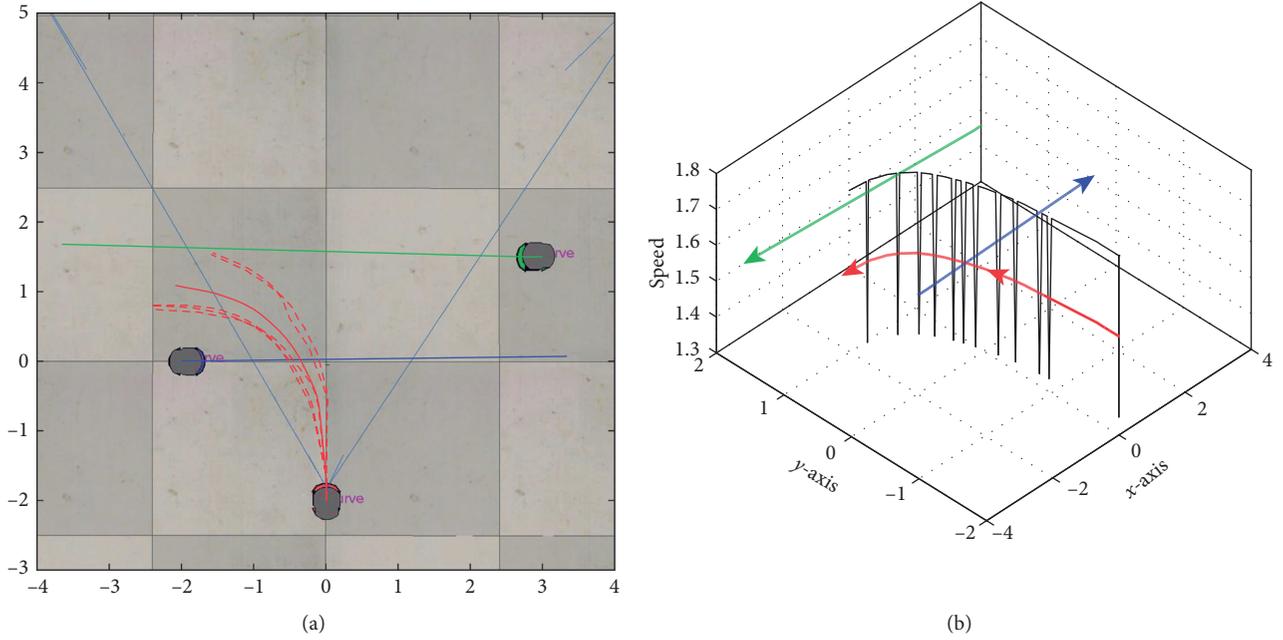


FIGURE 7: Output response using the real-time Hermite OF method. (a) Trajectory of the target robot: five attempts (dashed lines) and mean (red line); (b) speed representation of the target robot (dark line) over the trajectory.

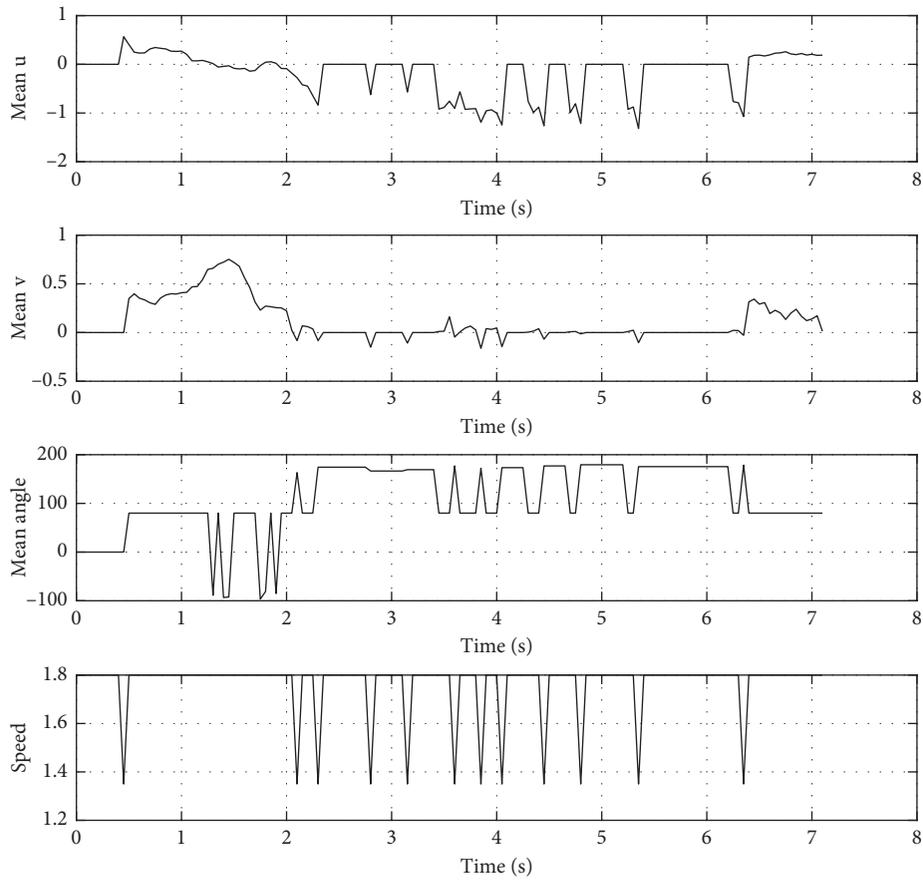


FIGURE 8: Inputs (\bar{u} , \bar{v} , $\bar{\theta}$) and output (*speed*) values for the controller.

```

(1) if  $\|\bar{u}\| > T_u$  and  $\bar{v} > T_v$  then
(2)   if  $\bar{\theta} < T_l$  or  $\bar{\theta} > T_u$  then
(3)     steer to the left
(4)   else if  $\bar{\theta} > T_{m_1}$  or  $\bar{\theta} < T_{m_2}$  then
(5)     steer to the right
(6)   else
(7)     go forward
(8)   end if
(9) end if
    
```

ALGORITHM 1: Set of rules for the first experiment used in obstacle avoidance.

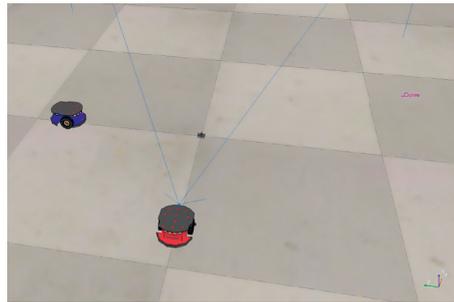


FIGURE 9: Scenario developed for testing avoidance of a mobile obstacle (blue robot).

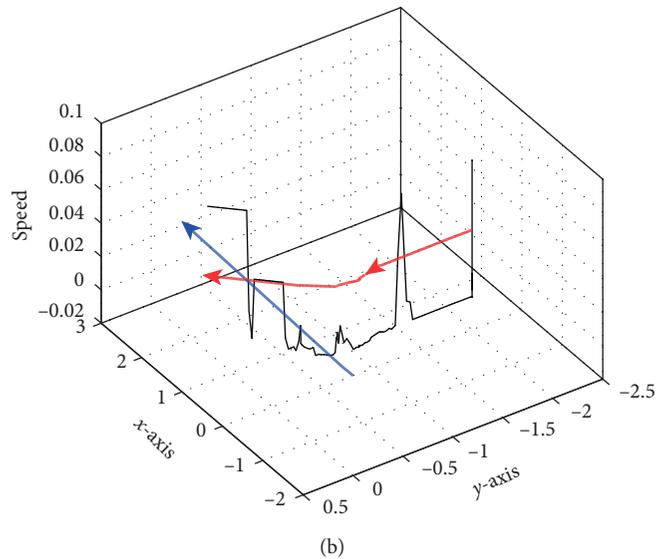
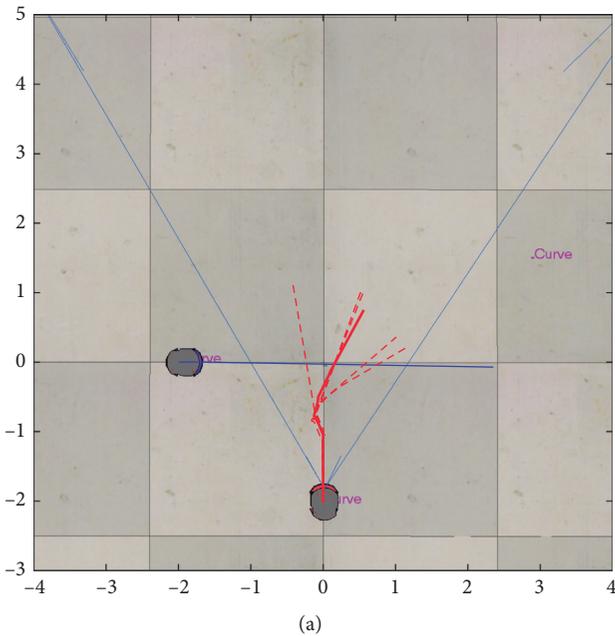


FIGURE 10: Output response using RT-HOF and AOC. (a) Trajectory of the target robot: five attempts (dashed lines) and mean (red line); (b) speed representation of the target robot (dark line) over the trajectory.

is the target distance measured from the sensor, \hat{d}_q is the estimated distance from the CNN, and N is the size of the testing data. Also, the accuracy was measured as the number of times the difference between the target and estimated

distances is below or equal to a threshold ϵ , as shown in Equation (11). After testing, the performance of the CNN with OF obtained 0.0591 in RMSE (meters) and 90.7% in accuracy with threshold $\epsilon = 0.10$ m.

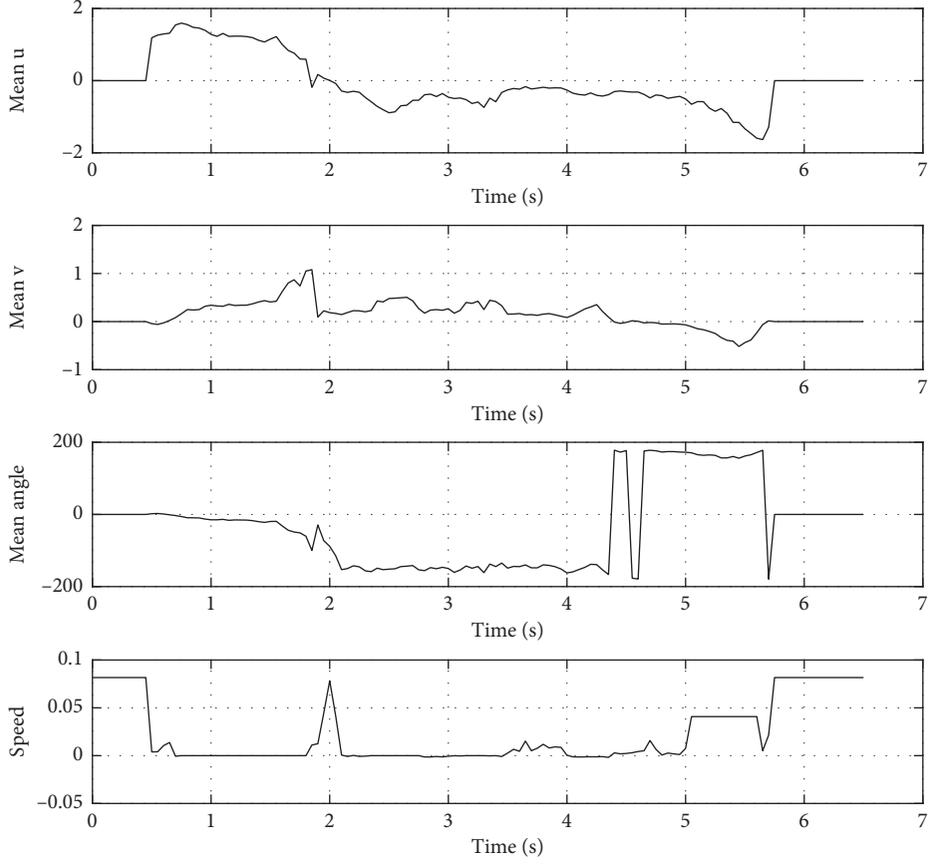


FIGURE 11: Inputs (\bar{u} , \bar{v} , $\bar{\theta}$) and output ($speed$) values for the controller.

TABLE 2: Security and smoothness indexes for the robot navigation using the real-time Hermite OF.

Attempt	SM1 (m)	SM2 (m)	SM3 (m)	TB _E	B _E	S _k
1	1.4796	1.3166	0.62518	4.6097e-05	7.8663e-08	1.0344e-06
2	1.4756	1.415	0.7146	4.518e-05	7.4678e-08	1.0005e-06
3	1.6943	1.2665	0.58068	4.9117e-05	7.5448e-08	9.6104e-07
4	1.7563	1.2261	0.54734	4.6815e-05	6.5844e-08	9.2914e-07
5	1.6145	1.364	0.66258	4.471e-05	7.5017e-08	9.8786e-07
Mean	1.6041	1.3176	0.62608	4.6384e-05	7.393e-08	9.8259e-07

$$RMSE = \sqrt{\frac{\sum_{q=1}^N (d_q - \hat{d}_q)^2}{N}}, \quad (10)$$

$$accuracy = \frac{\sum_{q=1}^N \{ |d_q - \hat{d}_q| \leq \epsilon \}}{N}. \quad (11)$$

Thus, in Figure 12, it is depicted the initial configuration of the environment. It considers one fixed obstacle (red cylinder). The red robot shown in the scene is the one with the CNN-based distance estimation model that is used during the whole time in the experiment.

The output trajectories when using the CNN-based distance estimation model are shown in Figure 13. Five attempts were run (reported as the dashed red lines), and the mean trajectory (strong red line) is also depicted

(Figure 13(a)). It also shows the velocity of the target robot over its trajectory (Figure 13(b)). As shown, the target robot goes forward until it detects the proximity of the obstacle by estimating the distance to it with the CNN model. Also, an implemented AOC determines the velocities of the wheels, aiming the robot steering to the left. Figure 14 reports the input (\bar{d}) and output ($speed$) values for the controller. The behavior of the red robot is very precise, and during this experiment it did not collide with the cylinder.

In terms of the security and smoothness indexes, Table 4 summarizes the results for each of the attempts and the mean performance. As shown in the indexes, the robot reaches close positions to the obstacle, with a minimum distance of 0.42 m securing minimal risks in the robot. Smoothness indexes prove minimal abrupt changes in the trajectories when dealing with a fixed obstacle.

TABLE 3: Security and smoothness indexes for the robot navigation using the RT-HOF and AOC.

Attempt	SM1 (m)	SM2 (m)	SM3 (m)	TB_E	B_E	S_k
1	0.7473	1.6509	0.7473	$9.7704e-05$	$1.84e-07$	$2.4774e-06$
2	0.74157	1.6427	0.74157	$7.2661e-05$	$1.5044e-07$	$1.4464e-06$
3	0.85645	1.5416	0.85645	0.00013544	$2.7307e-07$	$3.3304e-06$
4	0.76235	1.9285	0.76235	$7.6021e-05$	$1.5483e-07$	$1.8303e-06$
5	0.79618	1.6155	0.79618	0.00010674	$2.3408e-07$	$2.4665e-06$
Mean	0.78077	1.6758	0.78077	$9.7714e-05$	$1.9928e-07$	$2.3102e-06$

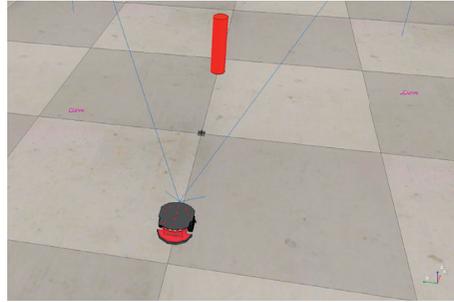


FIGURE 12: Scenario developed for testing avoidance of a fixed obstacle (cylinder).

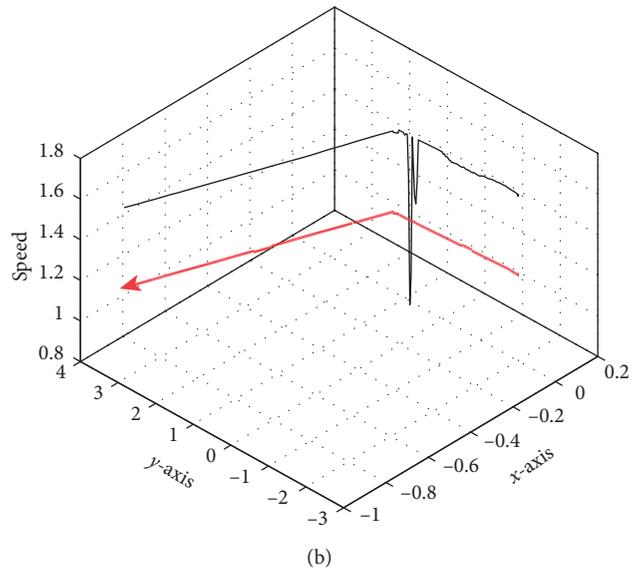
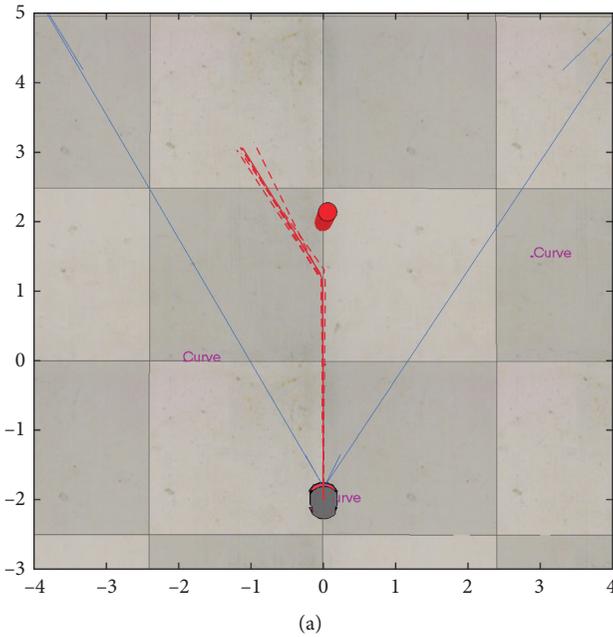


FIGURE 13: Output response using the CNN-based distance estimation method. (a) Trajectory of the target robot: five attempts (dashed lines) and mean (red line); (b) speed representation of the target robot (dark line) over the trajectory.

3.4. *Obstacle Avoidance and Free Navigation Using the Proposed Controller.* The last experiment aims to measure the output performance of the proposed nature-inspired control system for avoiding fixed and mobile obstacles and free navigating over the environment. Figure 15 shows the test scenario for this experiment. It comprises one fixed obstacle (red cylinder) and three mobile obstacles (green, blue, and yellow robots), as well as the target robot (red robot). The scene shown in Figure 15 represents the initial conditions. The target robot is the one with the proposed controller implemented as described in Section 2.5.

Figure 16 shows the output trajectories when using the proposed controller. Five attempts were run (reported in dashed red lines), and the mean trajectory (strong red line) is depicted in Figure 16(a), while Figure 16(b) shows the speed of the target robot over its trajectory. As noticed, the target robot dealt with both fixed and mobile obstacles. For instance, the first object seen by the robot is the cylinder. However, the blue robot is then crossed in front of the target robot. In this situation, the red robot decreases its velocity and steers to the left slightly. Once the blue robot is out of range, the target robot moves forward decreasing its velocity again while the

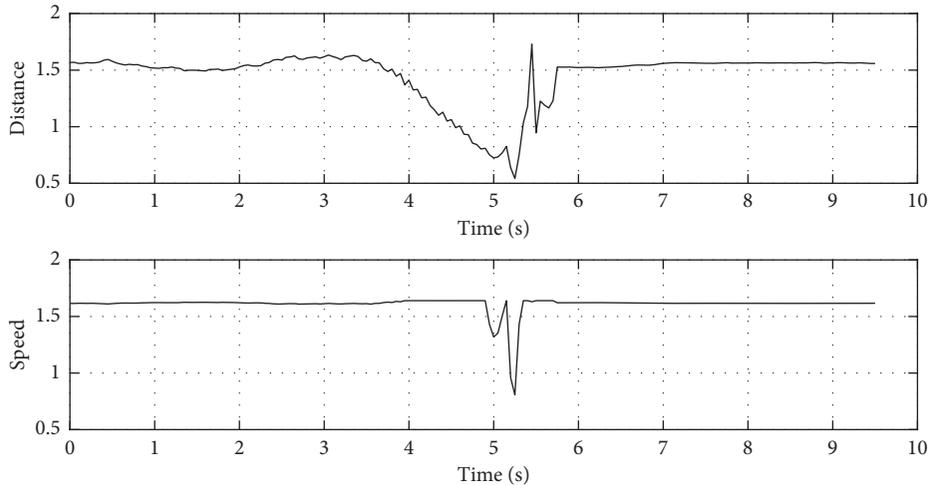


FIGURE 14: Input (\hat{d}) and output (*speed*) values for the controller.

TABLE 4: Security and smoothness indexes for the robot navigation using the CNN-based estimation method.

Attempt	SM1 (m)	SM2 (m)	SM3 (m)	TB_E	B_E	S_k
1	0.4241	1.9076	0.4241	$3.5541e-05$	$4.58e-08$	$5.5335e-07$
2	0.46384	1.9298	0.46384	$3.2683e-05$	$4.3577e-08$	$5.6792e-07$
3	0.37958	1.9096	0.37958	$3.7597e-05$	$4.9339e-08$	$6.4338e-07$
4	0.44578	1.9168	0.44578	$3.4012e-05$	$4.3162e-08$	$5.6113e-07$
5	0.37746	1.909	0.37746	$3.0383e-05$	$3.8754e-08$	$5.2752e-07$
Mean	0.41815	1.9146	0.41815	$3.4043e-05$	$4.4127e-08$	$5.7066e-07$

green robot is moving around. Later on, the green robot is out of the vision of the red robot, but the red cylinder is relatively closer. In that way, the target robot steers again to the left to avoid collision with the cylinder. At last, the yellow robot is not seen by the target robot, so it goes straight.

In addition, Figure 17 shows the inputs (\bar{u} , \bar{v} , \hat{d}) and output (*speed*) values of the proposed controller. Notice that the combined behavior of previous experiments is obtained in this proposal. For instance, the speed of the target robot decreases when \bar{u} is positive and \bar{v} is about zero (e.g., detecting a mobile obstacle), or when \hat{d} is small (e.g., detecting a fixed obstacle). The combination of \bar{u} and \bar{v} is correlated to the steering of the target robot.

We measure the security and smoothness of the trajectories, as shown in Table 5. It summarizes the results for each of the attempts and the mean performance. From these metrics, it can be observed that the robot maintained the minimal secure distance of 0.61 m from obstacles, reducing the risks for collisions. In terms of smoothness, this control system allowed the robot to navigate without larger changes in curvature.

It is remarkable to say that the vision of the robot might have more than one object at the time, increasing the difficulty for determining the best action of the robot. To this end, the target robot did not collide in any of the attempts conducted in this experiment, validating that the proposed controller can be implemented for avoiding obstacles and free navigating in scenarios where there are both fixed and mobile obstacles.

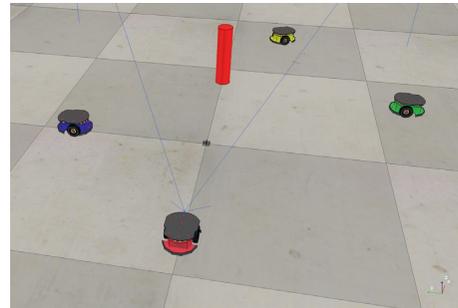


FIGURE 15: Scenario developed for testing obstacle avoidance and free navigation.

3.5. Discussion. In this work, we estimate the distance to the objects instead of a depth map of the scene as reported in several studies in the literature. For this task, we use a monocular camera to infer the distance and the object motion in contrast to the applications using stereo vision systems that are highly computational resource consuming. An advantage of our proposal is the use of two kinds of measures (motion and distance) that allow avoiding fixed and mobile obstacles. In addition, our method needs less data for training the distance estimator and no training data to compute the OF field, in contrast with the OF-based CNN approaches.

The use of these cognitive strategies complement the robot control, in contrast with traditional applications in robotic vision. Hence, it is the first time that CNN training

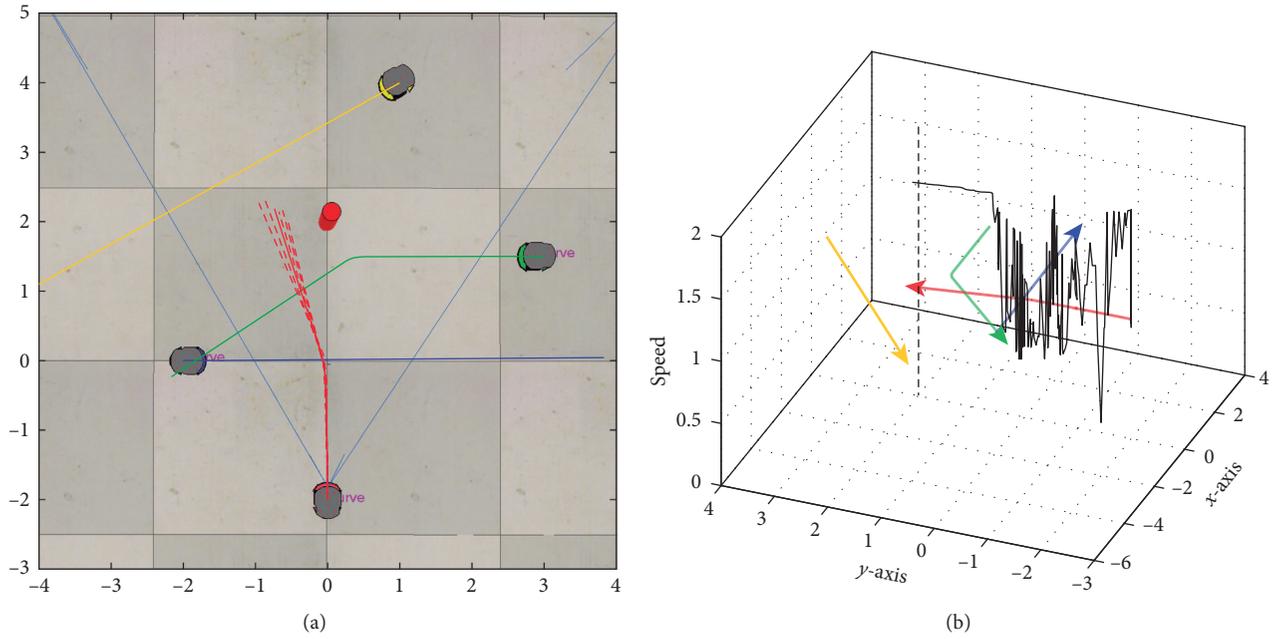


FIGURE 16: Output response using the proposed nature-inspired control system. (a) Trajectory of the target robot: five attempts (dashed lines) and mean (red line); (b) speed representation of the target robot (dark line) over the trajectory.

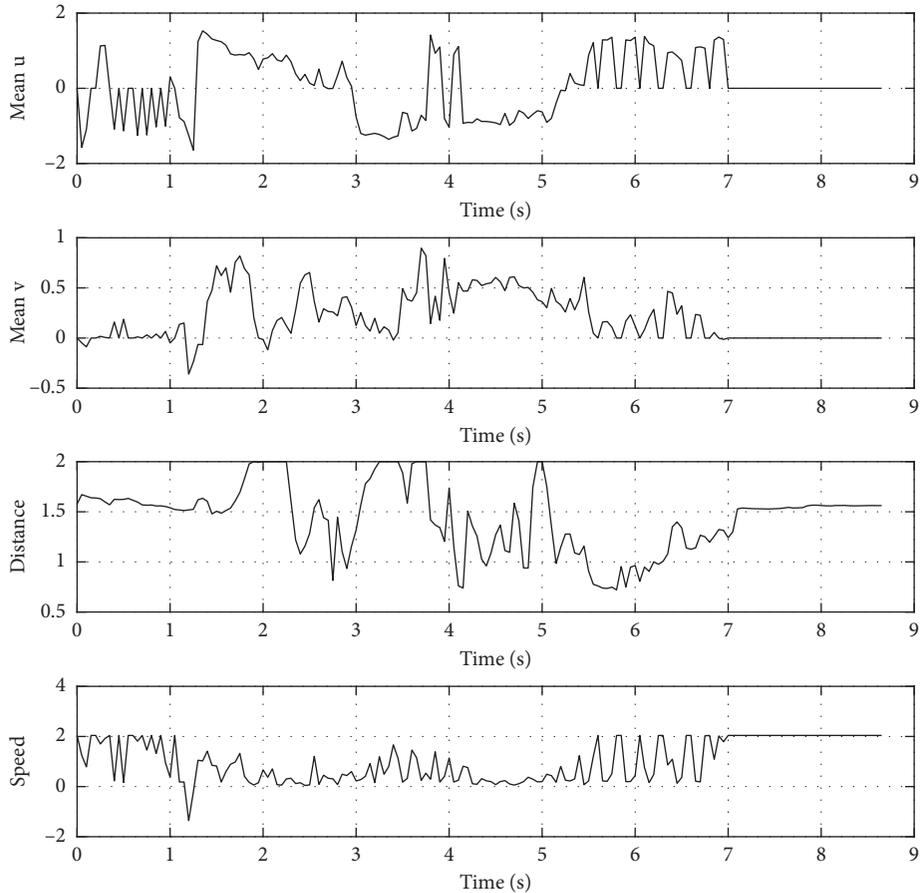


FIGURE 17: Inputs (\bar{u} , \bar{v} , \hat{d}) and output (*speed*) values for the proposed controller.

TABLE 5: Security and smoothness indexes for the robot navigation using the proposed nature-inspired control system.

Attempt	SM1 (m)	SM2 (m)	SM3 (m)	TB _E	B _E	S _k
1	1.1638	1.9244	0.6004	0.00079652	7.3548e-07	1.2218e-05
2	1.2589	2.0064	0.66768	0.00071704	6.3455e-07	9.8604e-06
3	1.1763	1.9168	0.56143	0.00089058	7.6973e-07	1.1408e-05
4	1.2612	1.9481	0.71119	0.00060311	5.3945e-07	9.062e-06
5	1.1768	1.9034	0.53153	0.00085002	7.7275e-07	1.2863e-05
Mean	1.2074	1.9398	0.61445	0.00077145	6.9039e-07	1.1082e-05

was carried out using ultrasound sensors. In addition, the AOC is implemented for handling uncertain information such as the estimations from the inputs, giving robustness to the control system. The control strategy uses the OF features directly without need of path planning over depth maps.

Some drawbacks of the proposed approach are the following: (1) this approach does not consider multiple objects in the scene, so complementary treatment would be required. (2) In the experiments, we use the robot simulator V-REP that recreates the conditions of the physical world quite accurately, but further investigation in real conditions are necessary.

4. Conclusions

In this paper, we presented an autonomous navigation system to be applied in RAL. We propose an integrated system including a vision sensor, real-time Hermite OF, and distance estimation by CNN to introduce to the intelligent robot controller based on AOC. We use the OF for motion estimation and CNN for distance inference to the objects, making this application suitable for avoiding fixed and mobile targets. Particularly, we use a monocular camera for the whole task.

We did four experiments to test different scenarios: using only OF and on-off control for mobile obstacles, a combination of OF and AOC, a CNN-based distance estimation and AOC for fixed obstacles, and the integration of OF, CNN-based distance estimation, and AOC. Simulation results were done in V-REP software and the results confirm that our approach is useful for obstacle avoidance and free navigation. In all test scenarios, there was never a collision with the objects using our proposal. A quantitative analysis was done using the security and smoothness metrics applied to the control navigation response. These quantitative metrics suggest that using the AOC strategy allows to avoid obstacles in a comfortable way and with minimal abrupt changes in the trajectory. To this end, the experiments confirm that a monocular camera can be applied for robot navigation tasks.

The proposed approach successfully combines a bio-inspired OF method, a CNN technique for distance inference, and a novel hybrid fuzzy logic and artificial hydrocarbon networks controller system. This integration loosely simulates a high cognitive vision strategy that allows analyzing holistic information from the egocentric point of view of the mobile robot.

As future work, we will test our approach over RAL in real scenarios in order to improve the navigation performance of mobile robots in presence of dynamic environments typically found in the context of elderly people places.

Data Availability

The dataset used in this work was collected by the authors, and it can be found in <http://sites.google.com/up.edu.mx/robotflow/>.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by Universidad Panamericana through the grant “Fomento a la Investigación UP 2018,” under project code UP-CI-2018-ING-MX-05.

References

- [1] U. Nations, *World Population Prospects 2017 Revision, Chapter Globally, Population Aged 60 or over is Growing Faster than All Younger Age Groups*, United Nations, New York City, NY, USA, 2017.
- [2] M. Shishehgar, D. Kerr, and J. Blake, “A systematic review of research into how robotic technology can help older people,” *Smart Health*, vol. 7-8, pp. 1–18, 2018.
- [3] X. B. Jin, T. L. Su, J. L. Kong, Y. T. Bai, B. B. Miao, and C. Dou, “State-of-the-art mobile intelligence: enabling robots to move like humans by estimating mobility with artificial intelligence,” *Applied Sciences*, vol. 8, no. 3, p. 379, 2018.
- [4] P. Pinggera, D. Pfeiffer, U. Franke, and R. Mester, *Know Your Limits: Accuracy of Long Range Stereoscopic Object Measurements in Practice*, Springer International Publishing, Cham, Switzerland, 2014.
- [5] M. Cordts, L. Schneider, M. Enzweiler, U. Franke, and S. Roth, *Object-Level Priors for Stixel Generation*, Springer International Publishing, Cham, Switzerland, 2014.
- [6] B. Shahraray and M. K. Brown, “Robust depth estimation from optical flow,” in *Proceedings of Second International Conference on Computer Vision*, pp. 641–650, Tampa, FL, USA, December 1988.
- [7] B. K. P. Horn and B. G. Schunck, “Determining optical flow,” *Artificial Intelligence*, vol. 17, no. 1–3, pp. 185–203, 1981.
- [8] S. Diamantas, A. Oikonomidis, and R. Crowder, “Depth computation using optical flow and least squares,” in *Proceedings of 2010 IEEE/SICE International Symposium on System Integration*, pp. 7–12, Sendai, Japan, December 2010.
- [9] S. Diamantas, A. Oikonomidis, and R. Crowder, “Depth estimation for autonomous robot navigation: a comparative approach,” in *Proceedings of 2010 IEEE International Conference on Imaging Systems and Techniques (IST)*, pp. 426–430, Thessaloniki, Greece, July 2010.

- [10] H. He, Y. Li, and J. Tan, "Relative motion estimation using visual-inertial optical flow," *Autonomous Robots*, vol. 42, no. 3, pp. 615–629, 2018.
- [11] J. Julin, K. Ravi Kumar, and S. Mahendran, "Optical flow based velocity estimation for vision based navigation of aircraft," *International Journal of Applied Engineering Research*, vol. 11, no. 6, pp. 4402–4405, 2016.
- [12] A. Popov, A. Miller, B. Miller, and K. Stepanyan, "Estimation of velocities via optical flow," in *Proceedings of 2016 International Conference on Robotics and Machine Vision*, vol. 10253, Moscow, Russia, February 2017.
- [13] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.
- [14] A. Mousavian, H. Pirsiavash, and J. Kosecka, "Joint semantic segmentation and depth estimation with deep convolutional networks," in *Proceedings of 2016 Fourth International Conference on 3D Vision (3DV)*, pp. 611–619, Stanford, CA, USA, October 2016.
- [15] U. K. Thikshaja, A. Paul, S. Rho, and D. Bhattacharjee, "An adaptive transcurvative algorithm for depth estimation in deep learning networks," in *Proceedings of 2016 International Conference on Platform Technology and Service (PlatCon)*, pp. 1–3, Jeju, Republic of Korea, February 2016.
- [16] P. Wang, W. Li, Z. Gao, J. Zhang, C. Tang, and P. O. Ogunbona, "Action recognition from depth maps using deep convolutional neural networks," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 4, pp. 498–509, Aug 2016.
- [17] M. Mancini, G. Costante, P. Valigi, and T. A. Ciarfuglia, "Fast robust monocular depth estimation for obstacle detection with fully convolutional networks," in *Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4296–4303, Daejeon, Republic of Korea, October 2016.
- [18] H. Ponce, J. Brieva, and E. Moya-Albor, "Distance estimation using a bio-inspired optical flow strategy applied to neuro-robotics," in *Proceedings of 2018 International Joint Conference on Neural Networks*, IEEE, Rio de Janeiro, RJ, Brazil, October 2018.
- [19] E. Rohmer, S. Singh, and M. Freese, "V-REP: a versatile and scalable robot simulation framework," in *Proceedings of 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, November 2013.
- [20] E. Moya-Albor, J. Brieva, and H. E. P. Espinosa, *Mobile Robot with Movement Detection Controlled by a Real-Time Optical Flow Hermite Transform*, Springer International Publishing, Cham, Switzerland, 2016.
- [21] H. Ponce, E. Moya-Albor, and J. Brieva, "A novel artificial organic controller with hermite optical flow feedback for mobile robot navigation," in *New Applications of Artificial Intelligence*, P. Ponce, A. M. Gutierrez, and J. Rodriguez, Eds., Chapter 6. InTech, Rijeka, Croatia, 2016.
- [22] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6602–6611, Honolulu, HI, USA, July 2017.
- [23] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5506–5514, Las Vegas, NV, USA, June 2016.
- [24] R. Garg, B. G. Vijay Kumar, G. Carneiro, and I. Reid, "Un-supervised CNN for single view depth estimation: geometry to the rescue," in *Computer Vision—ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., pp. 740–756, Springer International Publishing, Cham, Switzerland, 2016.
- [25] J. Peng, Z. Xiong, D. Liu, and X. Chen, "Unsupervised depth estimation from light field using a convolutional neural network," in *Proceedings of 2018 International Conference on 3D Vision (3DV)*, pp. 295–303, Verona, Italy, September 2018.
- [26] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1119–1127, Boston, MA, USA, June 2015.
- [27] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5162–5170, Boston, MA, USA, June 2015.
- [28] Y. Gan, X. Xu, W. Sun, and L. Lin, "Monocular depth estimation with affinity, vertical pooling, and label enhancement," in *Computer Vision—ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., pp. 232–247, Springer International Publishing, Cham, Switzerland, 2018.
- [29] X. Cheng, P. Wang, and R. Yang, "Depth estimation via affinity learned with convolutional spatial propagation network," in *Computer Vision—ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., pp. 108–125, Springer International Publishing, Cham, Switzerland, 2018.
- [30] Z. Chen, V. Badrinarayanan, G. Drozdov, and A. Rabinovich, "Estimating depth from RGB and sparse sensing," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., pp. 176–192, Springer International Publishing, Cham, Switzerland, 2018.
- [31] D. Ron, K. Duan, C. Ma et al., "Monocular depth estimation via deep structured models with ordinal constraints," in *Proceedings of 2018 International Conference on 3D Vision (3DV)*, pp. 570–577, Verona, Italy, September 2018.
- [32] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proceedings of 2016 Fourth International Conference on 3D Vision (3DV)*, pp. 239–248, Stanford, CA, USA, October 2016.
- [33] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Un-supervised learning of depth and ego-motion from video," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6612–6619, Honolulu, HI, USA, July 2017.
- [34] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proceedings of 27th International Conference on Neural Information Processing Systems (NIPS)*, Montreal, Canada, December 2014.
- [35] L. Pan, Y. Dai, M. Liu, and F. Porikli, "Simultaneous stereo video deblurring and scene flow estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6987–6996, Honolulu, HI, USA, July 2017.
- [36] R. Ranftl, V. Vineet, Q. Chen, and V. Koltun, "Dense monocular depth estimation in complex dynamic scenes," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4058–4066, Las Vegas, NV, USA, June 2016.
- [37] H. Jiang, G. Larsson, M. Maire, G. Shakhnarovich, and E. Learned-Miller, "Self-supervised relative depth learning for urban scene understanding," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss,

- Eds., pp. 20–37, Springer International Publishing, Cham, Switzerland, 2018.
- [38] K. Tateno, F. Tombari, I. Laina, and N. Navab, “CNN-SLAM: real-time dense monocular SLAM with learned depth prediction,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6565–6574, Honolulu, HI, USA, July 2017.
 - [39] B. Galvin, B. Mccane, K. Novins, D. Mason, and S. Mills, “Recovering motion fields: an evaluation of eight optical flow algorithms,” in *Proceedings of British Machine Vision Conference*, pp. 195–204, Southampton, UK, September 1998.
 - [40] J. B. Martens, “The hermite transform–theory,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, no. 9, pp. 1595–1606, 1990.
 - [41] J. Gu, Z. Wang, J. Kuen et al., “Recent advances in convolutional neural networks,” *Pattern Recognition*, vol. 77, pp. 354–377, 2018.
 - [42] K. Nogueira, O. A. B. Penatti, and J. A. dos Santos, “Towards better exploiting convolutional neural networks for remote sensing scene classification,” *Pattern Recognition*, vol. 61, pp. 539–556, 2017.
 - [43] H. Ponce, P. Ponce, and A. Molina, “Artificial hydrocarbon networks fuzzy inference system,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 531031, 13 pages, 2013.
 - [44] H. Ponce and P. Ponce, “Artificial organic networks,” in *Proceedings of 2011 IEEE Conference on Electronics, Robotics and Automotive Mechanics*, pp. 29–34, Cuernavaca, Morelos, Mexico, November 2011.
 - [45] H. Ponce, P. Ponce, and A. Molina, *Artificial Organic Networks: Artificial Intelligence Based on Carbon Networks, volume 521 of Studies in Computational Intelligence*, Springer, Berlin, Germany, 2014.
 - [46] H. Ponce, “A novel artificial hydrocarbon networks based value function approximation in hierarchical reinforcement learning,” in *Proceedings of 15th Mexican International Conference on Artificial Intelligence*, pp. 211–225, Cancún, Mexico, October 2016.
 - [47] H. Ponce, P. Ponce, and A. Molina, “The development of an artificial organic networks toolkit for LabVIEW,” *Journal of Computational Chemistry*, vol. 36, no. 7, pp. 478–492, 2015.
 - [48] N. D. M. Ceballos, J. A. Valencia, and N. L. Ospina, “Quantitative performance metrics for mobile robots navigation,” in *Mobile Robots Navigation*, A. Barrera, Ed., chapter 24, IntechOpen, Rijeka, Croatia, 2010.