UNICA

UNIVERSITÀ
DEGLI STUDI
DI CAGLIARI

Ph.D. DEGREE IN
Mathematics and Computer Science
Cycle XXXV

TITLE OF THE Ph.D. THESIS

Sensor-based artificial intelligence to support

people with cognitive and physical disorders

Scientific Disciplinary Sector

INF/01

Ph.D. Student: Silvia Maria Massa

Supervisor Prof. Daniele Riboni

Final exam. Academic Year 2021/2022
Thesis defence: April 2023 Session


Massa Silvia Maria gratefully acknowledges the Sardinian Regional Government for the financial support of her PhD scholarship (P.O.R. Sardegna F.S.E. - Operational Programme of the Autonomous Region of Sardinia, European Social Fund 2014-2020 - Axis III Education and training, Thematic goal 10, Investment Priority 10ii), Specific goal 10.5.

Abstract

A substantial portion of the world's population deals with disability. Many disabled people do not have equal access to healthcare, education, and employment opportunities, do not receive specific disability-related services, and experience exclusion from everyday life activities.

One way to face these issues is through the use of healthcare technologies. Unfortunately, there is a large amount of diverse and heterogeneous disabilities, which require ad-hoc and personalized solutions. Moreover, the design and implementation of effective and efficient technologies is a complex and expensive process involving challenging issues, including usability and acceptability.

The work presented in this thesis aims to improve the current state of technologies available to support people with disorders affecting the mind or the motor system by proposing the use of sensors coupled with signal processing methods and artificial intelligence algorithms.

The first part of the thesis focused on mental state monitoring. We investigated the application of a low-cost portable electroencephalography sensor and supervised learning methods to evaluate a person's attention. Indeed, the analysis of attention has several purposes, including the diagnosis and rehabilitation of children with attention-deficit/hyperactivity disorder. A novel dataset was collected from volunteers during an image annotation task, and used for the experimental evaluation using different machine learning techniques.

Then, in the second part of the thesis, we focused on addressing limitations related to motor disability. We introduced the use of graph neural networks to process high-density electromyography data for upper limbs amputees' movement/grasping intention recognition for enabling the use of robotic prostheses. High-density electromyography sensors can simultaneously acquire electromyography signals from different parts of the muscle, providing a large amount of spatio-temporal infor-

mation that needs to be properly exploited to improve recognition accuracy. The investigation of the approach was conducted using a recent real-world dataset consisting of electromyography signals collected from 20 volunteers while performing 65 different gestures.

In the final part of the thesis, we developed a prototype of a versatile interactive system that can be useful to people with different types of disabilities. The system can maintain a food diary for frail people with nutrition problems, such as people with neurocognitive diseases or frail elderly people, which may have difficulties due to forgetfulness or physical issues. The novel architecture automatically recognizes the preparation of food at home, in a privacy-preserving and unobtrusive way, exploiting air quality data acquired from a commercial sensor, statistical features extraction, and a deep neural network. A robotic system prototype is used to simplify the interaction with the inhabitant. For this work, a large dataset of annotated sensor data acquired over a period of 8 months from different individuals in different homes was collected.

Overall, the results achieved in the thesis are promising, and pave the way for several real-world implementations and future research directions.

Contents

Chapter 1

Introduction

According to the 2004 World Health Survey and Global Burden of Disease (GBD) estimates, and based on 2010 population estimates, there were around 785 (15.6%) - 975 (19.4%) million persons 15 years and older living with a moderate or severe disability. Of these, around 110 (2.2%) - 190 (3.8%) million experienced severe disability. If we also consider children, it is estimated that more than one billion people (or about 15% of the world's population) live with a disability [O+11]. However, most people, in their lives, will deal with some form of (at least temporary) disability. Disability is usually associated with a permanent situation, but a person might also temporarily experience disability, for instance, due to a broken bone. In addition, more and more non-disabled people are taking on the responsibility of supporting and caring for a relative or friend with disabilities.

This problem will worsen as the population increases. According to [oESA22], in 1950 the estimated global population was 2.5 billion, in 2022 it reached 8.0 billion, and in 2059 it is expected to exceed 10 billion. With advances in the medical field, older people are also increasing along with the rest of the population. In 1980 the estimated number of people over the age of 65 was 258 million, and in 2022 it reached 771 million. Moreover, the elderly population is expected to reach 994 million in 2030 and 1.6 billion in 2050. Unfortunately, global aging has a strong influence on disability trends. In GBD 2004 [O+08], we observe that 46.1% of persons 60 years or older live with a moderate and severe disability, while if we consider people aged 15 or older, the percentage decreases to 19.4%. If we examine only people with severe disabilities, we observe that the percentages are 10.2% (age 60 or older) and 3.8% (age 15 or older), respectively.

The International Classification of Functioning, Disability and Health (ICF), officially approved by all 191 WHO member states, is the international standard for describing and measuring health and disability at the individual and population levels [ICF]. According to ICF, disability arises from the interaction of health conditions, that are diseases, injuries, and disorders, with environmental and personal factors. Environmental factors include products and technology; the natural and built environment; support and relationships; attitudes; and services, systems, and policies. Personal factors include motivation and self-esteem which can influence how much a person participates in society.

The United Nations Convention on the Rights of Persons with Disabilities (CRPD) [CRP] also states that "disability is an evolving concept and that disability results from the interaction between persons with impairments and attitudinal and environmental barriers that hinder their full and effective participation in society on an equal basis with others."

In both definitions, disability is described as a consequence of interaction and not as an attribute of a person. Consequently, disability today is seen as a matter of more or less, not yes or no. For example, the lack of a sign language interpreter creates disabilities by creating barriers to participation and inclusion for deaf people that would otherwise not exist. The barriers that limit people with disabilities access to healthcare, education, and employment opportunities, ultimately leading to exclusion from activities of daily living, are mainly due to the absence of disability-specific services. The situation can be mitigated, for example, through legislation, policy changes or technological developments. Historically, people thought that the only way to assist a person with a disability was through solutions that segregated them, such as residential institutions and special schools. Today, however, the solution is thought to be improving social participation by addressing the barriers that hinder people with disabilities in their daily lives [O+11]. The process of lowering or removing barriers is complex, often different, and requires vision, skills, incentives, resources, and an action plan.

In this thesis, we see how barriers can be addressed and thus inclusiveness can be improved through the use of health technologies that leverage sensors and artificial intelligence algorithms. There are several types of disabilities, and each has specific health, educational, rehabilitation, social, and support needs. In addition, different responses may be needed because different people with the same disability may have

very different experiences and needs. For the sake of this thesis, our work focuses only on cognitive and physical disorders.

The thesis is organized as follows:

- In Chapter 2, we analyze existing solutions regarding: recognition of a person's mental state while performing an activity with the ultimate goal of diagnosing attention-deficit/hyperactivity disorder (ADHD) and helping children affected by it to improve their concentration; detecting the intention of upper limb amputees in order to move a prosthetic robotic hand; and maintaining a food diary, with the aim of providing an alternative solution for frail people with physical or memory problems who live alone at home. In particular, we focused on the sensors used in the literature and the problems that can be encountered with their use, such as high cost, intrusiveness, difficulty of use, and ineffectiveness in solving the problem being addressed.

- In Chapter 3, we investigate the use of electroencephalography (EEG) data and machine learning algorithms to monitor performance in attention-demanding tasks. To this aim, we present a system created to evaluate the performance of an image annotator by analyzing EEG signals acquired through a low-cost sensor. We explain the data collection method employed to acquire the new dataset created to experimentally evaluate the system. In particular, it is shown how the acquired EEG signal was divided into sliding windows of variable length, how features were extracted from each of them through the use of zero padding and discrete Fourier transform (DFT), and how feature vectors were later used to train and test the support vector machine (SVM) and random forest (RF) classifiers.

- In Chapter 4, we present a system created to assess a person's attention state during the execution of different tasks by analyzing EEG signals acquired through two different sensors. In this work we used two datasets: a public dataset, and the dataset collected during the work presented in Chapter 3. Compared to the previous work, we split the acquired brainwave signals using a different sliding windows approach, and we extracted different features from each window to train and test the RF classifier.

- In Chapter 5, we present a system created to recognize the movement intentions of people with upper limb amputations based on data collected through

high-density electromyography (HD-EMG) electrodes. We illustrate the public dataset used, the graph neural network (GNN) created to analyze the HD-EMG data, and finally, we explain how the graphs used to train and test the GNN are created based on the structure of the electrodes used to acquire the data.

- In Chapter 6, we present a system which, through the acquisition of data by means of an air quality sensor, is able to automatically recognize meals cooking activities and, with the support of a robotic assistant, helps the inhabitant to interactively keep a food diary. We acquired a large real-world dataset from several volunteers to experimentally evaluate the system. In that Chapter, we illustrate the hardware/software architecture, the designed neural network for cooking activity recognition, the technique for acquiring and processing the data, the feature extraction method, and the achieved results.

- We present our conclusions in Chapter 7. We discuss the problems that we encountered during the course of our works, and the solutions that we envision for addressing challenging research issues that remain open.

–

Chapter 2

Literature Review

The objective of this Chapter is to highlight the various problems related to the use of specific sensors in the context of certain challenges, such as monitoring mental state, the movement of a robotic prosthesis, or the maintenance of a food diary, and to propose possible solutions.

Table 2.1 summarises the characteristics of all the sensors illustrated in the following Chapter.

In particular, in Section 2.1 we discuss EEG sensors use, in Section 2.2 we discuss EMG sensors use, and in Section 2.3 we discuss the different sensors currently used in food journaling systems and the possible use of an air quality sensors as an alternative acquisition method in these systems.

Table 2.1: Summary of sensors illustrated in Chapter 2

Attention state monitoring	
camera	- effective - invasive in terms of privacy
EEG sensor	- channels number, effectiveness, cost, and fixation system complexity are variable and closely related (generally, when one of these increases, the others also increase and vice versa) - privacy-friendly
Body movement intention recognition	
IMU sensor	- can only be used to recognize gestures in combination with EMG
ENG sensor	- restore the sense of touch and alleviate phantom limb symptoms - ineffective in recognising the amputee's intention
EMG sensor	- most widely used sensor to date - each electrode contains only one channel (limited number of channels that can be used to collect signals)
HD-EMG sensor	- each electrode contains a large number of channels (e.g. 64) positioned on a matrix (acquisition of a large amount of spatial information)
Food diary maintance system	
text	- require effort by users in the long term
camera / camera + other sensors	- invasive in terms of privacy - obtrusive - expensive
air quality sensor + microphone	- privacy-friendly - unobtrusive - inexpensive

2.1 Mental state monitoring

The first part of the thesis focused on attention state monitoring, which has several applications. Wargnier et al. present a user interface based on embodied conversational agents (ECA) to assist older adults with cognitive impairment. The system aims to compensate for attention disorders by prompting the elderly person to regain attention whenever a state of inattention is detected. Attention is monitored by tracking the user's posture and facial orientation since typically during an interaction one stands in front of the interlocutor [WMJ+15]. In [VHS10] through the monitoring of visual attention using an eye tracker and the use of some questionnaires, the authors manage to verify the factors that seem to influence consumers' attention to nutritional information on food products. Similarly, the reasons that influence a person to read health warnings on cigarette packs can be verified in [MRBL11]. Batista proposed a system to make driving safer by measuring the driver's attention levels through camera-acquired data related to principal facial features points, blinking, and eye rotation and closure [Bat07]. In [LZX+13], a robotic assistant is presented that adjusts the position of the laparoscope when performing procedures based on the surgeon's visual attention, which is monitored by following the surgeon's eye movements via an eye tracker. Canedo et al. propose an autonomous agent that can monitor a classroom through the analysis of a series of photos in which the attention state is monitored for each student through facial features and body pose tracking [CTN18].

In these works, attention status is monitored using cameras, which can often be perceived by several people as very invasive in terms of privacy. For this reason, in our works, the data needed to monitor the subject's attention status are collected through the use of an EEG sensor, that allows neurophysiological responses to be recorded [LGSMV14].

The existing works regarding the use of EEG to monitor attention are mainly related to ADHD [Fur05]. ADHD is one of the most common mental disorders. Its symptoms are not the same for all children, and they can be: the inability to concentrate; short attention spans; vulnerability to external interferences; poor inhibition; difficulty controlling emotions and behaviors; and easy impulsivity. As a result, children with ADHD face various difficulties. For instance, the inability to complete homework independently, the inability to focus on listening in class, and

obstacles in language expression and reading comprehension [O+19].

Monitoring attention while playing serious games for ADHD (SGAD) has been proposed to solve problems associated with traditional methods of diagnosis and treatment. Indeed, to detect the presence of ADHD, clinicians usually use questionnaires, statistical manuals, or structured interviews to assess patients' daily behaviors. These methods of diagnosis are often not objective, and they can be influenced by the evaluator's opinions and the unnatural behaviors of young people in front of doctors and boring questionnaires. In addition, ADHD cannot be treated so that it ceases totally, so once the diagnosis is made, stimulant medications are used to relieve symptoms. Unfortunately, these medications can have side effects or lead to addiction [ZLL+21].

A large number of experiments have shown that SGAD can effectively distinguish ADHD children from non-ADHD children, as well as reduce ADHD children's symptoms and improve executive functions [PCJLGS+20].

In [ASEE18] Alchalabi et al. present an SGAD, called FOCUS, which is played using EEG signals acquired through the EMOTIV EPOC+ kit shown in Figure 2.1. FOCUS helps the subjects to train and reinforce attention by challenging them to move an avatar to collect the largest possible number of cubes in the shortest possible time while concentrating and using mental commands.

Thanasuan et al. develop an SGAD, called Armis, that provides neurofeedback for adults' attention training. The game character, which the player controls via the keyboard, wakes up on the "bridge of hell" and must survive attacks until the end of the game level. The attention state is measured through the NeuroSky MindWave BCI system shown in Figure 2.1 and is displayed on the game user interface. When the player's attention level is low, the background of the game slowly darkens, leaving the field of view only in the center of the screen to help the player focus on a small area of the game and regain attention. Each level requires more concentration to pass [OLKT17].

The game character in the SGDA proposed in [AP15] is a superhero academy student who must improve his superpower, which is attention, to Fight supervillains. The game includes a 3D school environment, active 3D distractions, and 2D games performed on the blackboard. The distractions are realistic such as paper planes on the blackboard, children running in the hallway, and the teacher walking in the classroom, and their goal is to make the player lose points. The SGDA is controlled by

steady-state visual evoked potential (SSVEP) registered through three gold-plated electrodes placed on the subject's scalp at the Oz (signal), Fpz (ground) and Fz or A1 (reference) positions according to the 10-20 electrode placement system.

In [BMLG16] another SGDA was proposed, called Harvest Challenge, which provides neurofeedback by monitoring attention during the course of the game using the NeuroSky MindWave BCI system shown in Figure 2.1. The player is asked to perform three different activities to train different types of attention, such as selective and sustained attention. The first proposed task, which must be accomplished within 5 minutes, is to collect from a panel presented on the screen the necessary equipment for a safe ride in the canopy (a helmet, a pair of gloves, a harness, and sports shoes) using the left and right keys of the keyboard. In the second task, the player is asked to collect bananas, apples, and pears and to repair the wooden ladder, that is used to reach the top of the mountain where the canopy rope is located, by increasing the level of attention. Finally, in the last task, the player is in a field and has to collect as many carrots as possible within 30 minutes while keeping the attention level high and constant, in fact, when a low attention level is detected the carrots hide under the ground and cannot be collected.

Machado et al. propose an SGDA in which the difficulty increases level by level. The player's first task is to drive a spaceship and collect the stars it encounters on the way. In the next level, the player must keep an eye on the level of gasoline in the tank. When this runs out, the spaceship keeps moving but cannot collect other stars until the player picks up some fuel. Eventually, a penalty is added for every second the spaceship tank is out of gasoline (-1 point). For each task, the subject has 5 minutes, and one point is earned for each star collected. Attention is monitored by EEG headset Quick-20 shown in Figure 2.1, and its level is communicated through the speed of the spaceship. In fact, the higher the player concentration, the higher the speed of the spaceship [MCFdR19].

Several issues must be considered when using an EEG system, such as effectiveness, cost, and fixation system. Usually, in work where EEG signals are used as input, artificial intelligence algorithms are exploited to identify the subject's mental state or intention. For an artificial intelligence model to be effective, it is essential that it receives as input all the information needed to decode EEG signals correctly. This is why a large number of electrodes are generally used in systems. The greater the number of electrodes used, the greater the amount of temporal and spatial in-

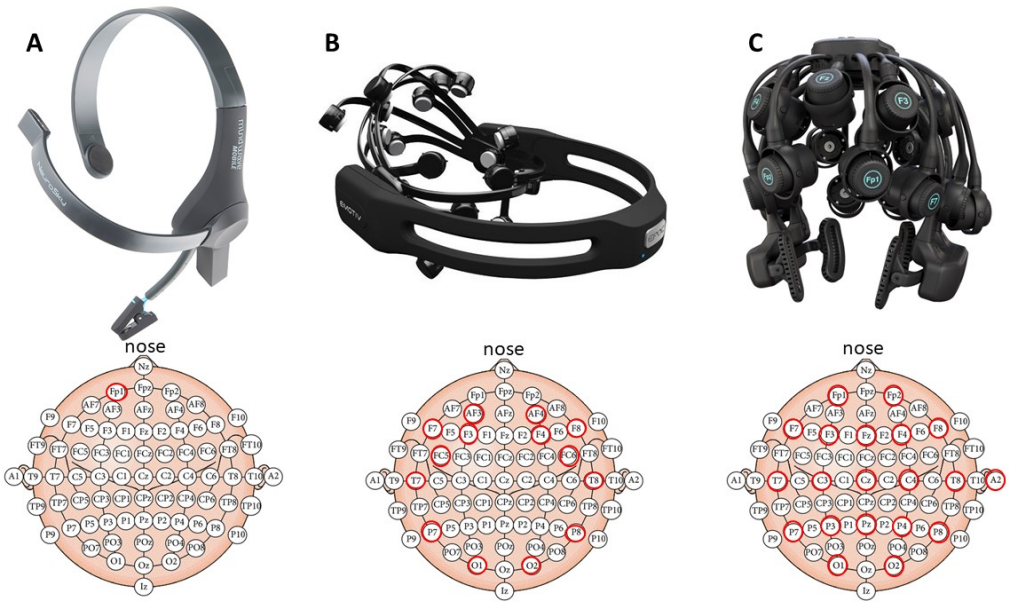

Figure 2.1: From left to right, NeuroSky MindWave BCI system (A), EMOTIV EPOC+ kit (B), Quick-20 EEG headset manufactured by CGX (C). Below, in red, for each sensor we show the position of the respective electrodes on the scalp according to the 10-20 electrode placement system. A modified version of (B) was used in the work presented in Chapter 4.

formation available. However, a greater number of electrodes results in higher costs. In addition, electrodes must be placed on the scalp following the international 10-20 system [HHP87]. This task is not easy when electrodes have to be placed one by one, especially for unskilled people, For this reason, in recent times, dry EEG electrodes [LGSMV14] attached to headsets and headbands have been proposed to make the EEG electrodes easier and faster to place on the scalp.

These problems are present in the works listed so far where EEGs are used. In fact, [ASEE18, MCFdR19] used headsets containing respectively 14 and 20 electrodes. Unfortunately, these headsets can be purchased on the market at a high price, thus making the use of the presented SGDA not accessible to everyone. In [OLKT17, BMLG16] is used an headset that contains only one electrode. Compared to the previous one this is a low-cost sensor, however, the presence of a single electrode makes the acquired data insufficient to optimally identify the attention level. Finally, the work presented in [AP15] used three electrodes, that need to be placed on the scalp one at a time in the positions Oz, Fpz, and Fz or A1. Of course, to

do this one must have a good knowledge of the 10-20 electrode placement system, which does not make the use of SGDA executable outside a hospital setting.

Therefore in the first part of the thesis, we decided to use in our works the low-cost portable EEG sensor Muse headband version 2 of InteraXon [mus] shown in Figure 2.2. This sensor is sold at an affordable price for families, is capable of simultaneously acquiring EEG signals from 4 electrodes placed in TP9, AF7, AF8, and TP10 according to the 10-20 electrode placement system, and is easy to apply due to its headband structure where the electrodes are mounted.

In the work presented in Chapter 4, we compare the use of data collected with the Muse with the use of data collected with the Epoc sensor (using only 7 of the 14 available electrodes) to assess a person's attention state. Demonstrating that the use of a low-cost sensor does not compromise the effectiveness of the final system.

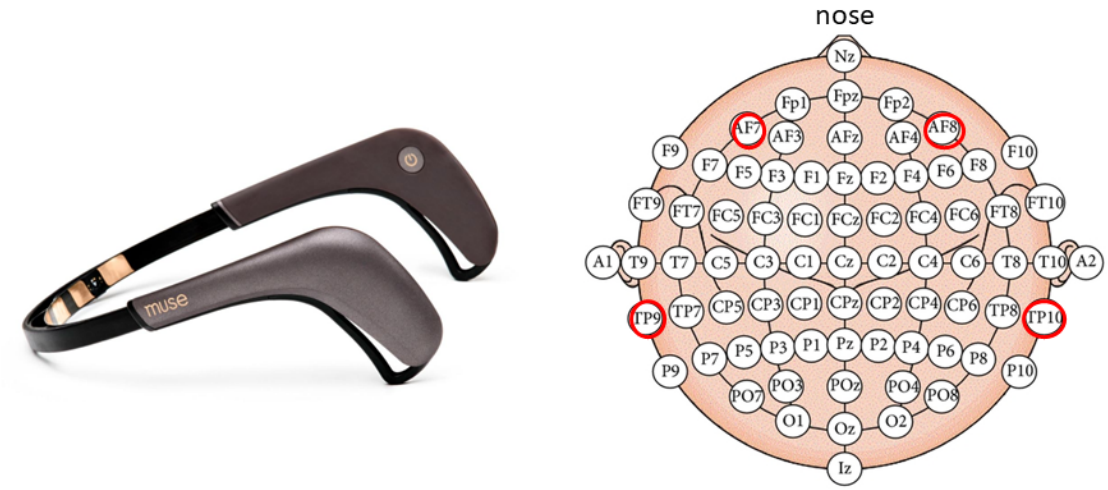

Figure 2.2: On the left, the figure shows the low-cost portable EEG sensor Muse headband version 2 of InteraXon used in the works presented in Chapter 3 and Chapter 4. In red, on the right, we show the position of the Muse's electrodes on the scalp according to the 10-20 electrode placement system.

The EEG data acquired with the Muse sensor and supervised learning were used in the work presented in Chapter 3 to automatically evaluate the performance of humans performing image annotation tasks.

An automatic method for evaluating the labeler performance may be useful to help children with ADHD to train their concentration state thanks to sound and visual neurofeedback provided in real-time regarding annotation correctness and

labeling speed. Figure 2.3 shows a mockup of a possible SGDA in which is used the automatic method for evaluating the labeler performance presented in this thesis.

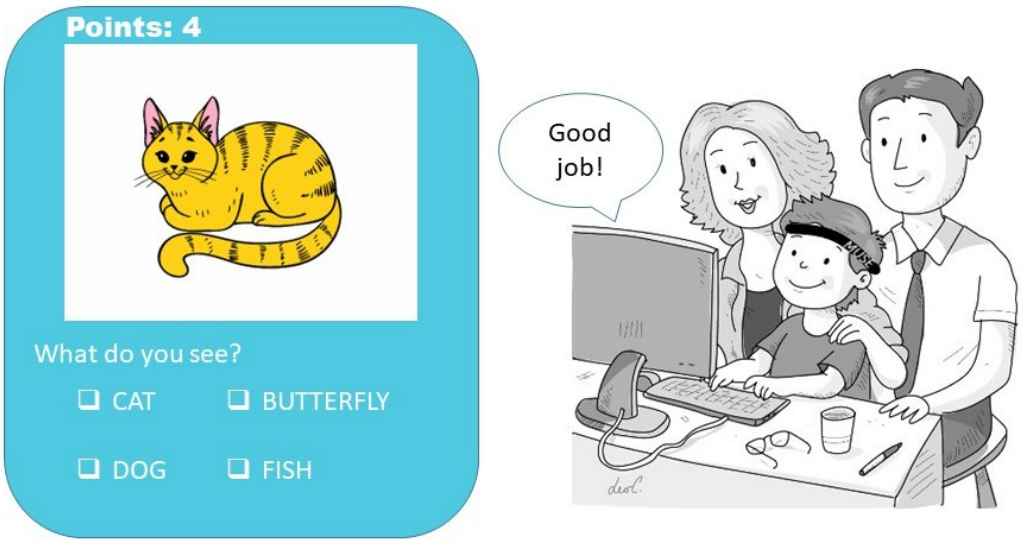

Figure 2.3: Mockup of a possible SGDA in which the labeler's automatic performance evaluation method is used to train concentration state. The player wears the Muse sensor. Familiar and easy to recognize images are presented to the child. For each image, a time is given to provide the answer and a point is awarded each time the answer is correct. Visual feedback is provided showing the number of points earned during the course of the game. In addition, auditory feedback is provided based on the correctness of the label and the time taken to give it.

It can also be useful for estimating the quality of annotations and detecting worker stress or fatigue. In fact, because of the increasing use of artificial intelligence methods for solving different tasks, there is an increasing need for large, well-annotated datasets. This is because many effective artificial intelligence algorithms require labeled data for model training. This is even more evident when using deep neural networks, which need very large training datasets to achieve unprecedented recognition rates in different domains [CL14].

Of course, in order to be effective, artificial intelligence algorithms need accurate annotations. Since most labels are manually set by domain experts, the current process of data annotation is relatively slow and costly, and prone to annotation errors. At the time of writing, most methods for evaluating the quality of annotations rely on interannotator agreement [Art17]. With this method, the same object is labeled by different annotators, and the level of agreement among annotators is considered.

Obviously, this process introduces redundancy and slows down the production of annotated datasets. Hence, there is increasing interest in devising advanced AI-supported tools for expediting the annotation task and for automatically evaluating the performance of the annotator, discovering possible labeling errors.

The usage of EEG in the data annotation domain is not new. In a previous work, Healy et al. used EEG data to study the role of attention and perception in an image annotation task [HGS16]. Parekh et al. proposed an artificial intelligence method to automatically annotate images based on EEG signals collected from an observer [PSRJ17]. However, to the best of our knowledge, the work we will present in more detail in Chapter 3 is the first research effort that applies the EEG data mining approach to evaluate the annotator performance. Indeed, while a few works investigated the use of EEG data for assessing the level of human attention [JJ20], [HZG+09], in our work we focus on the labeling performance instead of computing a generic attention level.

Usually, the process followed to quantify human cognitive performance according to EEG signals comprises [IK20]:

- A first data cleaning phase to remove artefacts, which is generally performed by applying a series of filters;
- A second phase of feature extraction from each signal. The most commonly used method is the Fast Fourier transform (FFT) which allows us to move from the time domain to the frequency domain and to obtain information regarding the different brainwaves.
- A final phase in which AI algorithms are used to classify the EEG signals features. The support vector machine (SVM) classifier was the most widely employed.

2.2 Body movement intention recognition

The second part of the thesis focused on using EMG signals acquired with surface electrodes from residual stump muscles to control robotic hand prostheses [JYBMM20, PSB+19].

Upper limb amputation severely limits daily activities performance such as grasping and manipulating objects or communicating. The leading cause of this condition

is trauma (80%), which occurs mainly in people aged 15-45 years. The second most prevalent are cancer or tumors and vascular complications of diseases. Upper limb amputation can be more or less severe depending on the level at which it is performed (Figure 2.4). Trans-phalangeal amputations are the most common among upper limb amputations (78%) [MA19]. It is clear that to restore use through a robotic prosthesis using EMG signals it is necessary to safeguard the muscles as much as possible.

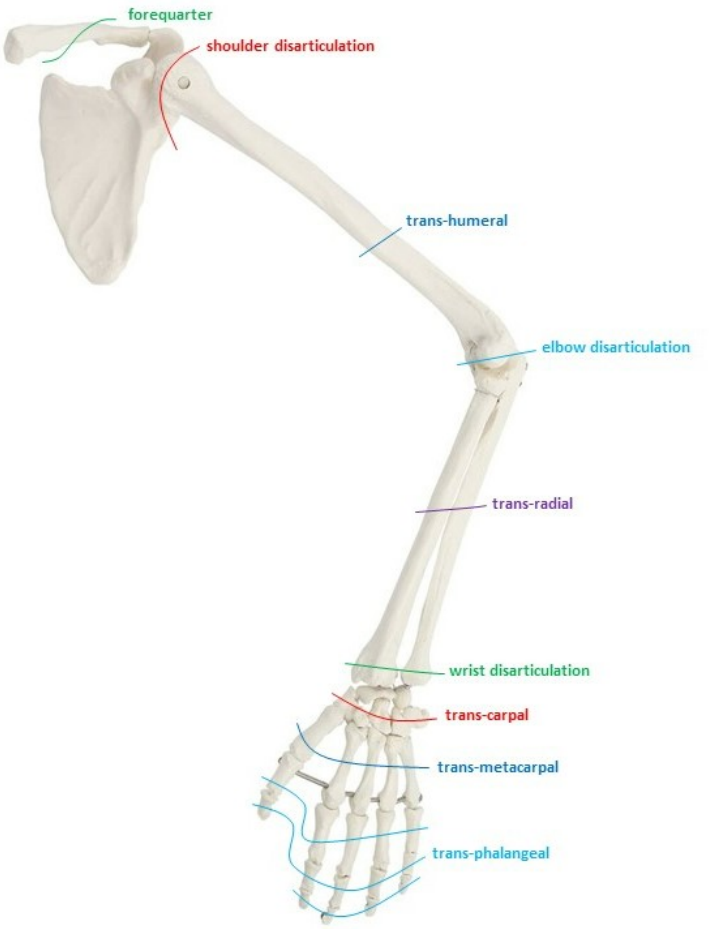

Figure 2.4: The different levels at which an amputation affecting an upper limb can be performed. The greater the portion of the arm that is lost, the more difficult it will be to restore its use through a robotic prosthesis.

For millennia, humans have worked on designing devices to facilitate the reintegration into society of people with a hand amputation [ZO14, Foo20, Put05].

Two hand prostheses for cosmetic use dating back to 2000 BC [Fri72] and 200

BC [RSNB94] have been found in Egypt.

In the seventh book of Natural History, published in 77 AD by the Roman scholar Pliny the Elder, the first prosthetic hand used for war purposes is mentioned. Especially, Pliny tells the story of the Roman general Marcus Sergius, who lost his right hand during the Second Punic War (218-201 BC) and later returned to battle wearing an iron prosthetic hand [Bea05].

Historical, archaeological, or iconographic evidence of prosthetic hands before the 16th century is rare. The main reasons appear to be the lack of knowledge to manage the hemorrhages and infections from amputations, which often led to death, and the fact that only the wealthy could afford such devices.

Among the most famous ancient prostheses is the iron hand created for the German knight Götz von Berlichingen to return to battle. The prosthesis that was equipped with five fingers that could be flexed and extended passively and locked was built after Götz lost his hand during the siege of Landshut (circa 1505) in Bavaria [Ott21].

The earliest evidence of prosthetic hands for non-war use is dated around 1600 and is attributed to the Italian surgeon Giovanni Tommaso Minadoi, who during a trip to Asia saw two men with upper limb amputations able to remove their hats, open a bag and hold a pen thanks to a prosthesis [Put05].

In 1818, the dentist Peter Baliff made the first body-powered upper limb prosthesis. It could be moved by intact muscles of the trunk and scapular girdle and leather straps capable of transmitting tension [Foo20]. The prosthesis was no longer conceived as a separate, foreign object usable only with the support of the other hand. Body-powered prostheses have been widely used since their creation, despite the presence of some problems such as fatigue related to their long-term use, inability to perform complex motor tasks, and nonhuman appearance.

In 1960, the first clinically significant myoelectric prosthesis was presented by Russian scientist Alexander Kobrinski [She64], and since then numerous studies on myoelectric prostheses have been conducted. Nowadays, almost all robotic hand prostheses use signals acquired by surface EMG electrodes.

EMG signals allow the acquisition of information regarding the electrical activity of skeletal muscles during the performance of certain tasks [TBC+20].

The use of EMG can be useful for different purposes, such as assessing muscle fatigue [RVCM20], evaluating abnormal patterns of muscle disorder [GSS+19], stroke

and nerve injury in the upper limb [SHP+18], sign language recognition [SS16, SS15], rehabilitation [FHW+20], and device control [AIK+09] like prostheses [SLT+18], drones [DN21], input devices for a computer [AIK+09, Whe03], etc. In some of these cases, other types of sensors such as gloves [EJBS17], vision sensors [PS15], and inertial measurement units (IMUs) [MFE+16] can be used to detect gestures. However, these sensors have a limitation compared to EMG sensors. In fact, EMG sensors are one of the few sensors that can be used by amputees because not only they can collect data on the execution of a hand movement, but they can also identify the intention of the movement. The only other sensor listed above used in amputee gesture recognition is the IMU sensor, but only in conjunction with EMG sensors [JSZ+22].

Another type of body signal used to recognize gestures is the electroneurography (ENG) signal. These signals alleviate phantom limb symptoms and restore the sense of touch while recognizing the amputee's intention [MCR10]. However, the sensors used to collect this type of signal are less effective and more invasive than EMG sensors. The main reasons are that the electrode inserted transversely in the nerve moves and collects signals from different nerve fibers over time and surgery is required for placement.

As we mentioned at the beginning we concentrated on the use of EMG signals to control robotic hand prostheses. Most of the work conducted so far in this field has used a low number of EMG electrodes typically ranging from 2 to 16 [KN21]. However, the performance of machine learning systems is affected by the availability of spatial and temporal information [KKAJN18].

The spatial information refers to the EMG data collected from multiple muscle sites on the forearm, i.e. the number of electrode channels used. The temporal information consists of the length of the analysis window and the degree of window overlap. Smith et al. [SHLK10] observed that spatial and temporal information are directly related and when spatial information is increased, temporal information can be reduced without significantly reducing classification accuracy. Menon et al. in [MDCL+17] affirm that an increase in channels does not always allow the use of shorter-length windows. Moreover, they affirm that the benefits of adding more electrode channels are determined by the type of limb deficit and that increasing the number of electrodes does not result in uniform improvement in performance. When choosing the length of the analysis window, it is necessary to find a trade-off between

the classification error rate and the delay of the classifier [SHLK10]. The greater the length of the window, the greater the accuracy achieved, but the greater the length of the window, the greater the delay in the classifier's decision. Conversely, the shorter the window length, the shorter the delay in the classifier's decision, but the shorter the window length, the lower the accuracy achieved. The degree of window overlap, as opposed to window length, is not responsible for the change in the classification error rate [MDCL+17]. However, there are few works available that deal with temporal information and its interaction with spatial information, and researchers tend to decide the number of electrodes and the length of the window empirically.

To increase the possibility of extracting spatial information, HD-EMG electrodes have been proposed [HZLK08, SNF14]. These employ a large two-dimensional (2D) array of closely spaced channels to acquire a large number of signals simultaneously from different parts of the muscle [DSvEZ06]. The total number of channels proposed for HD-EMG ranges from 32, 128, 192, 256 and over 350 electrodes [KN21]. Figure 2.5 shows the HD-EMG electrode employed to collect the data used in the work presented in Chapter 5.

Furthermore, it has been observed that gestures that include more degrees of freedom are more easily recognized if the number of channels used is increased [LSK10]. The use of HD-EMG electrodes, therefore, seems promising for improving the usability of existing robotic hand prostheses and has been used to control robotic hands in some previous work including [DAFRLG18, JR19]. For this reason, we decided to employ data acquired through HD-EMG electrodes in the work shown in Chapter 5.

Typically, the process followed to recognize the amputee's intention based on EMG signals to move a robotic hand prosthesis includes [PSB+19]:

- An initial pre-processing phase used to eliminate signal acquisition noise, electromagnetic disturbances, signal instability, and motion artifacts due to electrodes and cables.

- A data segmentation phase using sliding windows that can be adjacent or overlapping.

- A feature extraction phase in the time domain (TD), in the frequency domain (FD), and/or time-frequency domain (TFD). The best results are obtained

using TD features, in particular, the most used to date are MAV, WL, ZC, SSC proposed by [HALI20]. Sometimes this phase is followed by the application of a dimensionality reduction (DR) technique.

- A classification phase where an AI algorithm is used to recognize the gesture. The most commonly used classification methods are linear discriminate analysis (LDA), support vector machine (SVM), multi-layer perceptron (MLP), and artificial neural network (ANN).

- A final and optional post-processing phase.

To decode the intention of the amputee, we decided to exploit a GNN. GNNs are helpful in a context where a high number of temporally-correlated spatial information is available [WST+22]. This kind of neural network is composed of several propagation modules, which propagate information between nodes so that the aggregated information can capture both feature-based and topological information [ZCH+20]. To our knowledge, no previous studies have used a GNN, in conjunction with HD-EMG signals, to identify the movement/grasping that the amputee intends to perform.

2.3 Food diary maintenance system

In the last part of the thesis, we present an automated, non-intrusive, privacy-friendly, and low-cost system that supports frail and disabled people in collecting data regarding diet. Diet data analysis is extremely important to evaluate the healthiness of an individual's nutrition and for setting up interventions when necessary.

The World Health Organization (WHO) maintains that malnutrition, meaning deficiencies, excesses, or imbalances in energy and/or nutrient in a person's energy and nutrient intake, affects every country in the world. Around 1.9 billion adults worldwide are overweight, while 462 million are underweight. Following an improper diet can be a health risk because it leads to the occurrence of serious diseases. In particular, overweight and obesity may cause cardiovascular diseases, diabetes, musculoskeletal disorders, and some cancer, while undernutrition may cause wasting (low weight-for-height), stunting (low height-for-age), underweight (low weight-for-age),

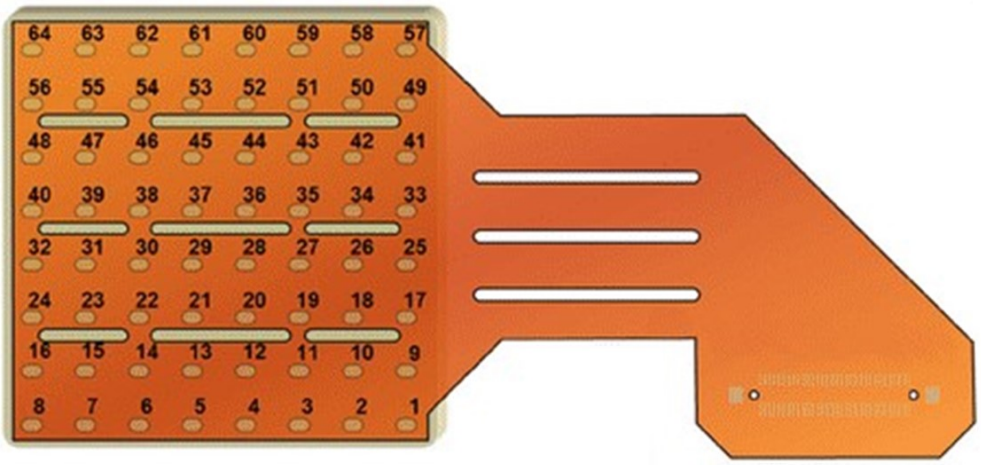

Figure 2.5: The HD-sEMG electrode used to collect the data for the work presented in Chapter 5. The electrode called ELSCH064NM3 and made by the OT Bioelectronics company contains 64 channels arranged in an 8x8 array with intrachannel spacing of 10mm.

and micronutrient deficiencies (a lack of vitamins and minerals) [WHO]. Moreover, underweight and higher levels of obesity have been associated with increased mortality compared to the normal-weight category [FGWG05].

Despite all the efforts of health education to improve eating habits, good dietary practices are often neglected, and there is a growing need for new technologies that can assist individuals in following good dietary practices [BHK+05].

A powerful tool for monitoring nutrition behavior is a food diary; i.e., a daily list of food taken by the individual, together with portion information. Accurate food journaling can support both self-management of nutrition routines [LLZL18], and assessment by practitioners [DHACN15]. In particular, for diabetes patients, it is important to analyse the daily food intake and compare it with symptom progression [MMK06]. A four-center randomized trial about weight loss maintenance shows that patients compiling a food diary lost twice as much weight than patients that kept no journaling [HGS+08]. Moreover, in the short term, food journaling may increase real-time awareness and mindfulness, avoiding the consumption of unhealthy food [WG07].

Traditional methods for keeping food diaries are based on interviews and questionnaires to assess the patients' eating routines [Mar71]. Manually keeping a constant journal of meals is considered tedious and time-consuming by many individuals, and food diaries are rarely taken accurately in the long term [CET+15]. However, there are several solutions for keeping independently a food diary that can reduce the burden of data acquisition [BRJ17]. These generally consist of mobile apps that frequently use smartphone sensors, such as the camera or the microphone [CBCF15, ZBW+10, ASLT05, MHH+02], and store locally or on the cloud food information and calorie counts for long-term monitoring [CBCF15]. But in the literature, we find solutions that also employ other types of hardware to acquire the data.

Casas et al. proposed an elementary text-based conversational agent [CMK18]. Zhu et al. employ computer vision tools and pictures taken before and after food consumption to accurately recognize the kind of food and estimate the eaten quantity [ZBW+10]. Sen et al. [SSM+18] created a smartwatch-based system to detect eating gestures, and recognize food through pictures and computer vision software. Chi et al. [CCCL08], provide accurate calorie counts using a combination of cameras, connected kitchen scales, and food databases. Yordanova et al. use data acquired from various sensors in a smart kitchen to identify cooking activities, particularly data regarding temperature, humidity, light/noise/dust levels, individual movements, use of certain objects, water, and electricity [YLW+19, YWP+17]. Other solutions rely on the automatic classification of chewing sounds [ASLT05], recognition of eating moments through analysis of heart rate and activity patterns [ONSJ18], or scanning of grocery receipts [MHH+02].

These solutions are not privacy-friendly, are obtrusive, expensive, and require effort by users in the long term [CET+15], especially if the users are elderly and suffering from forgetfulness or physical problems. In the work presented in Chapter 6, we address these issues by using a low-cost air quality sensor to automatically recognize food preparation at home.

Figure 2.6 shows the uHoo air quality sensor [uHo] with which the data are acquired in our work. It is a commercial sensor that does not require calibration or manual settings and can monitor several parameters, including temperature, humidity, carbon dioxide, volatile organic compounds, particulate matter, nitrogen dioxide, carbon monoxide, and ozone.

Figure 2.6: The Uhoo commercial air quality sensor used in the work presented in Chapter 6. Compared to other commercial air quality sensors, it can monitor a wide range of parameters, including temperature, humidity, carbon dioxide, volatile organic compounds, particulate matter, nitrogen dioxide, carbon monoxide, and ozone. In addition, it is simple to use because it does not require calibration or manual settings.

Air quality sensors are widely applied in the healthcare domain. Low-cost ones [RWJ+21, AK21, AIR+15, AKA20] are particularly popular because, compared to reference-grade air quality monitors, the purchase and operating costs are lower, the spatial density is higher, the acquired data can be displayed with different time-resolutions, and field distribution, data collection, and transmission are easier to implement [ZS20].

More and more of these low-cost air quality sensors are available in the market. Their characteristics are not standard and vary from sensor to sensor. These are commonly reported by the manufacturer in the sensor manual and include the following features: general operation such as charging mode, data storage and retrieval mode, operating conditions, possible expiration date, calibration mode, performance (accuracy and bias), maintenance mode, the response time when there is a change in conditions, pollutants detected, and known interference. Therefore, before buying

an air quality sensor, we need to ask ourselves several questions, such as: What do we want to measure? What is the sensor's ability to be accurate when it is far from the gas source or when the gas concentration is very low or very high? What are the accuracy and bias of the measurements? Is calibration necessary, and how is it done? What is the response time? What is the quality and durability of the hardware? Is the sensor usable for end users? How much does it cost? [WKS+14].

The sensors produced can monitor one or more air quality parameters, the most common are: ozone (O3), carbon monoxide (CO), carbon dioxide (CO2), sulfur dioxide (SO2), nitrogen dioxide (NO2), particulate matter (PM), volatile organic compound (VOC), temperature, and humidity. The possible sources of the mentioned gases are varied and often unknown, although we deal with them daily. Some of these are: electric utilities; gasoline vapors; unventilated fuel and gas-type space heaters; tobacco smoke; gas-type water heaters; wood stoves and fireplaces; gas-powered equipment; worn or poorly-adjusted and maintained combustion devices; people's breath; burning of fossil fuels by means of transport; cows farming; production of rice or other fruit and vegetable cultivation; combustion of coal, oil, and gas that contains sulfur [ZS20, gre].

When the air quality parameters assume abnormal values, various health disorders can arise such as fatigue; dizziness; nausea; eye, nose, and throat irritation; headache; flu-like symptoms; airway inflammation; respiratory disease; airway narrowing; chest pain; angina; reduced brain function; impaired vision and coordination; various degrees of toxic symptoms; lung infections; vascular and endothelial dysfunction; alterations in heart rate variability; coagulation; liver, kidney, and central nervous system damage; cancer; and fetal death [ZS20, WKS+14].

As it can be seen, the effects associated with exposure to polluted air can lead to consequences that vary in severity and include death [O+16]. Moreover, air pollutants can impact our lives by damaging vegetation, reducing visibility, and affecting global climate conditions [WKS+14]. It is, therefore, necessary to monitor air quality, especially for population groups that are most vulnerable to air pollution and most prone to develop a disease or an abnormal condition. These groups include: children aged 13 years or younger, the elderly aged 65 years or older, young people aged 18 years or younger with asthma, normal adults with asthma, and people with chronic obstructive pulmonary disease (COPD) [fDCC+94].

The most common purpose for which these sensors are employed is to provide

information on air healthiness [CYH+18, RTMH18]. However, the data acquired through these types of sensors can be used for more complex tasks, such as alerting a person when a specific event occurs and providing detailed information about a detected problem [JBR+17, SIA+15, PCP+21, IRT20].

In our work, the air quality sensor is used to automatically detect food preparation activities, with the goal of helping frail people living alone to keep a food diary. To our knowledge, this is the first time an air quality sensor has been used for this purpose. Therefore, there is no previous work that can give us information on which classifier is better to use to solve our problem. In this case, the procedure generally followed to choose the most suitable classifier consists of running tests with different classification algorithms and then comparing the final results obtained. The most tested classification algorithms in the literature are decision trees, Bayesian classifiers, SVMs and MLPs [BMJZOV17].

Chapter 3

EEG-based Performance Assessment in Attention-Demanding Tasks

As explained before, the ability of monitoring attention has several applications in healthcare for supporting frail people. Hence, in the work presented in this Chapter, we investigated the use of EEG data and machine learning for performance assessment in attention-demanding tasks.

We have collected a dataset from five volunteers carrying out an image annotation task. We used a window approach to compute feature vectors from the annotations and EEG data, and zero padding and discrete Fourier transform (DFT) [Sun23] to convert the data from the time domain to the frequency domain. Finally, we classified the windows using a supervised machine learning algorithm.

Section 3.1 describes the dataset that we have acquired. Section 3.2 explains our methodology. Section 3.3 reports our experimental evaluation.

3.1 Dataset collection

In this work, we collected a new dataset, acquiring EEG signals from people who were performing an image labeling task.

As mentioned earlier in Section 2.1, for the EEG data collection, we used the Muse headband version 2 of InteraXon [mus]. Muse consists of 4 electrodes that can collect information on brain activity with 256Hz sampling frequency in a non-

invasive way. The Muse electrodes collect signals from channels TP9, AF7, AF8, TP10, and Fpz, where the latter is only the reference electrode and does not capture brain signals. These electrodes are named and positioned according to the International System 10-20 [HHP87].

We used the mobile application Mind Monitor [min] along with the Muse sensor for receiving the EEG signals. Among the different data it returns, those used in our experiments are:

- Date and time of the recording.
- Raw brainwaves for each of the four sensors.
- Brainwaves Delta, Theta, Alpha, Beta, Gamma for each sensor.

The Raw EEG values represent the raw data of each sensor in microvolts, whose range goes from 0 µV to ∼1682 µV. The brainwave values are absolute band powers, based on the logarithm of the spectral power density (PSD) of the EEG data for each channel. These values are calculated internally by the Mind Monitor application with a data rate of 10Hz. The extracted brainwaves and their frequency bands are Delta 1-4Hz, Theta 4-8Hz, Alpha 7.5-13Hz, Beta 13-30Hz, and Gamma 30-44Hz [T+02]. Delta waves are related to deep sleep, unconsciousness, anesthesia, and lack of oxygen; Theta waves activity occurs when a person experiences emotional pressure, unconsciousness, or deep physical relaxation; Alpha waves are instead visible when an individual is in a state of consciousness, stillness, or rest, whereas when one is thinking, blinking or otherwise stimulated, this wave type disappears (alpha block); Beta waves are evident when a person thinks or receives sensory stimulation; finally, Gamma waves are related to selective attention, to the cognition, and perceptual activity [LCC13]. Also, Mind Monitor can recognize jaw clench and blinks.

Five volunteers (3 women, and 2 men, aged 24 to 42 years) with normal or correct vision and no known neurological damage participated in the data collection task. The participant's task was to label the indoor images that appeared randomly in a data annotation interface by selecting one of the eight buttons with the correct label. The labeling application was developed in Python by using the library Tkinter, and it is shown in Figure 3.2. The 8 labels in the interface are: 'computerroom', 'movietheater', 'library', 'kitchen', 'bowling', 'poolinside', 'trainstation',

'greenhouse'. The indoor images used are a subset of the dataset created in [QT09] which contains 15620 images divided into 67 indoor scenes, collected from various sources including online image search tools (Google and Altavista), online photo-sharing sites (Flickr), and the LabelMe dataset. At first, the application shows an image. As soon as the labeler presses a button, the application displays the next image. If the user does not press any button for 7 seconds, the label 'none' is associated with that image, and the next image is displayed. If the user responds within 7 seconds, the image is associated with the label selected by the user, and the next image is displayed. The application keeps track of the time taken by the annotator to choose the label.

In order to familiarize with the interface, each participant used the interface for 10 minutes without data acquisition. Next, he or she performed the actual image annotation task for about 30 minutes. During each annotation session, each user labeled approximately 1,000 images. The annotation tasks were all performed at the end of the day to reproduce a situation of tiredness. Before executing the task, annotators were told to avoid the displacement of the Muse headband not moving the head. Figure 3.1 shows a user during the labeling task.

After data collection, the EEG data collected through the Mind Monitor application and the data concerning the image labeling were aligned to create a single dataset. The resulting dataset has the following fields:

- Timestamp.
- Raw brainwaves for each of the four sensors at that timestamp.
- Brainwaves Delta, Theta, Alpha, Beta, Gamma for each of the four sensors.
- Correct image label.
- User assigned image label.
- Time taken by the user to annotate the image. If the time is 7 seconds, it means that the image was not annotated.

Finally, the class 'attentive' or 'distracted' has been assigned to the set of samples of each image. The labeling task is simple; therefore, we assume that most of the errors and the annotator's slowness are due only to distraction. Then if the label given by the annotator was wrong, the class assigned was 'distracted'. Otherwise,

Figure 3.1: A user wearing the Muse headband during the labeling task.

if the annotation was correct, we checked the response time. If the response time exceeded a certain threshold, the assigned class was 'distracted'; otherwise, it was 'attentive'. The response time threshold was set as 1.988 seconds, which is the average response time of all volunteers during the first 5 minutes of the experiment in which we assume they were concentrated and not tired.

After preprocessing, the final dataset has the following fields:

- Timestamp.
- Raw brainwaves for each of the four sensors.
- Brainwaves Delta, Theta, Alpha, Beta, Gamma for each of the four sensors.
- Time taken by the user to annotate the image.
- Assigned class ('distracted' or 'attentive').

After the label assignment, 3314 images were labeled as 'attentive' and 1280 as 'distracted'.

Figure 3.2: The interface used to perform the image labeling task. The randomly displayed images to be labeled are indoor images, such as computer room, movie theater, library, kitchen, bowling, pool inside, train station, and greenhouse. The labeler has 7 seconds to select a label among the 8 available. If a label is selected, it is associated with the image, and the next image is shown. On the other hand, if the labeler has not selected a label within 7 seconds, the label "none" is associated with the image, and the next image is shown.

3.2 Methodology

In this Section, we illustrate the main steps used to classify EEG signals for recognizing the attention state of the annotator. Figure 3.3 shows a flow chart illustrating our methodology.

3.2.1 Data cleaning and pre-processing

During data collection, the correct positioning of the Muse headband was verified through the Mind Monitor graphical user interface shown in Figure 3.4. Every time the sensor is not receiving a strong enough signal from one or more sensors, the

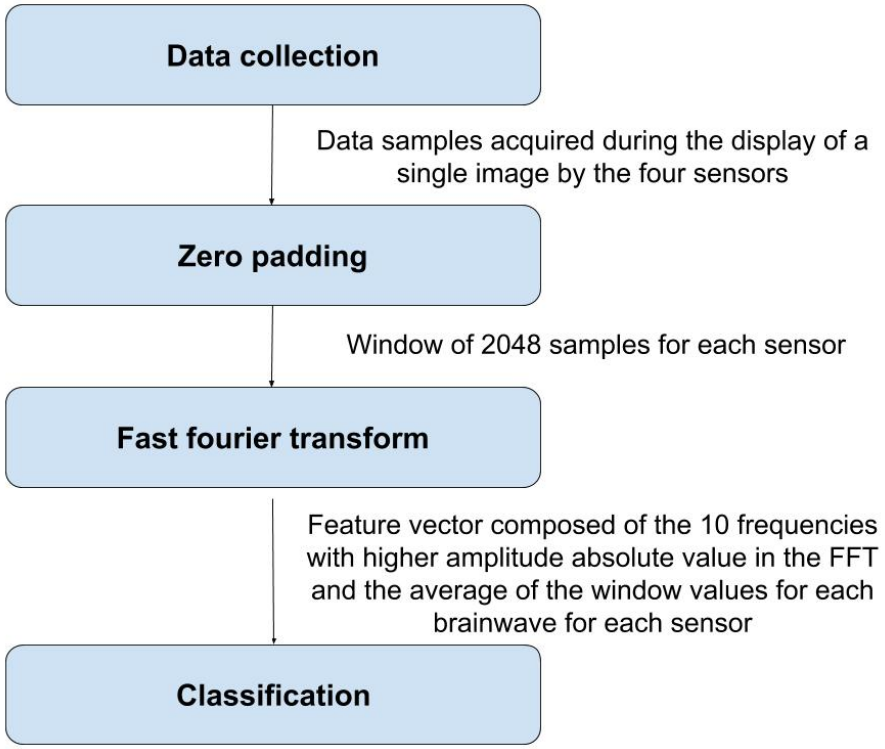

Figure 3.3: Flow chart of our methodology.

interface shows a white horseshoe. In fact, inside the horseshoe in Mind Monitor's interface, the Muse electrodes are represented by ovals that are solid if the connection is optimal or empty if the connection is good. Muse is a sensitive device and requires good skin contact to detect brain signals. The EEG signal may therefore have incorrectly recorded values or missing values in the recording. The quality of the recorded data has been improved by (i) removing samples with missing values concerning one or more channels, and (ii) disregarding those image annotations having more than 5% missing EEG samples.

The dataset was subsequently divided into windows, in which every window corresponds to the data samples acquired during the display of a single image. In order to have fixed-length windows, we set the window size 2048 samples. That value corresponds to 8 seconds, which is enough to represent all the data of every image labeling. Moreover, this dimension is powers of two and consequently improves the computation of the DFT [Sun23].

In order to reach the dimension of 2048, we used the zero padding technique which is performed by inserting a pad of zeros at the end of the sliding window.

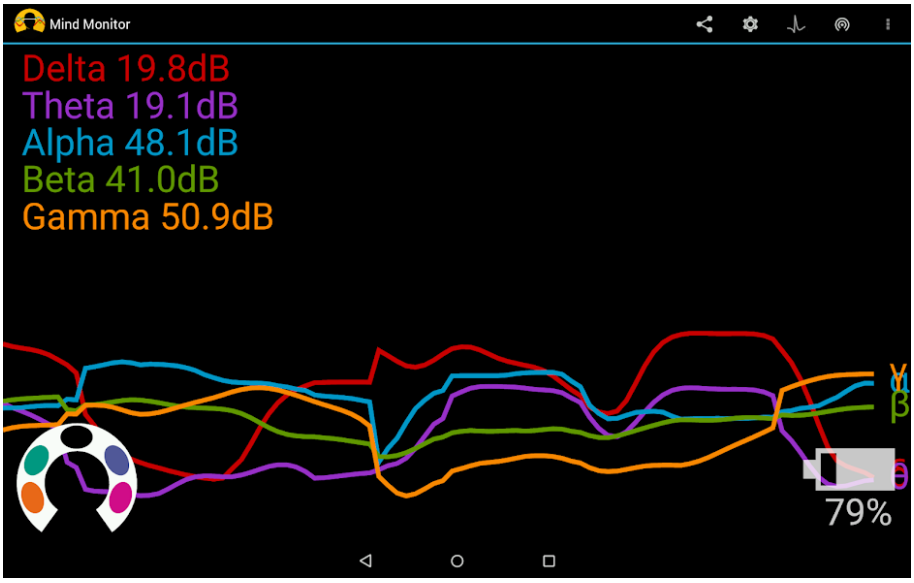

Figure 3.4: The Mind Monitor graphical interface. It allows the user to keep track of several aspects during the EEG data acquisition, such as the percentage of battery left to the Muse sensor and the brainwave values in real time. Another relevant aspect that can be monitored is the connection status of the Muse sensor electrodes to the skin. This is shown through the horseshoe, representing the headband, located in the lower left corner. Solid ovals represent a good connection, outlines represent a poor connection, and a blank space means no connection.

This technique is appropriate when the signal is limited in time and only interpolates between the frequency bins that would occur when no zero-padding is applied. However, zero-padding does not increase the frequency resolution of the DFT. Resolution is determined by the number of samples and sampling rate. Figure 3.5, Figure 3.6, and Figure 3.7 taken a sliding window show the raw signal, the signal after FFT application, and the signal after zero padding and FFT application, respectively.

3.2.2 Feature extraction

The DFT function converts a signal from the time domain into the frequency domain representation and has been calculated on each window using the Fast Fourier Transform (FFT) algorithm [CT65].

The FFT applied to the signal returns a vector of complex numbers of length N, dependent on the length L of the signal:

- $N = \frac{L}{2} + 1$ if L is even

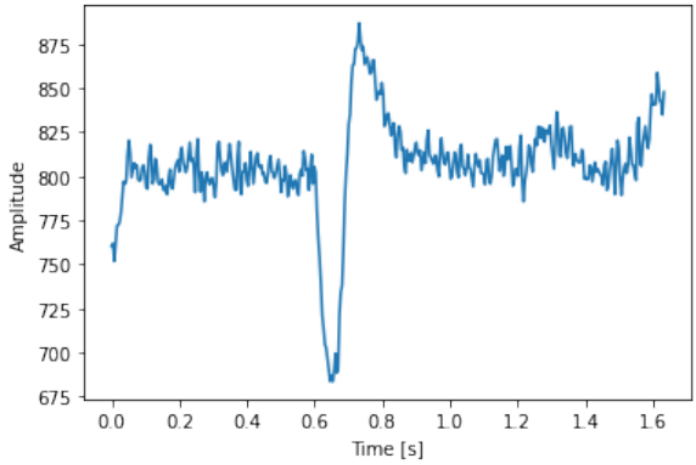

Figure 3.5: Plot of the raw signal acquired from the TP9 electrode collected during the viewing and labeling of an image.

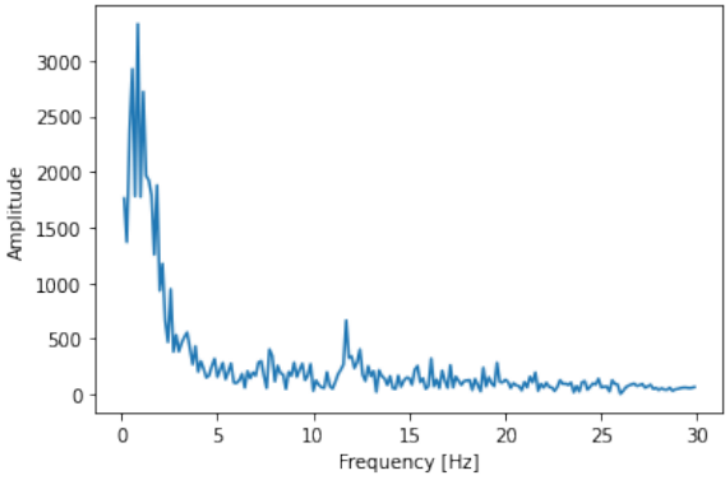

Figure 3.6: Plot of the signal in Figure 3.5 after application of the FFT.

- $N = \frac{L+1}{2}$ if L is odd

This vector contains the values of frequencies ranging from 0 to $fs/2$ Hz, where fs is the sampling frequency. Frequencies are represented with a step equal to $N = \frac{fs}{L}$. Therefore, in our case we have that L = 1025, and the frequencies represented range from 0 to 128 Hz with a step of 0.125 Hz.

The DFT allows us to inspect the trend of the different brainwaves that are identified by different frequencies.

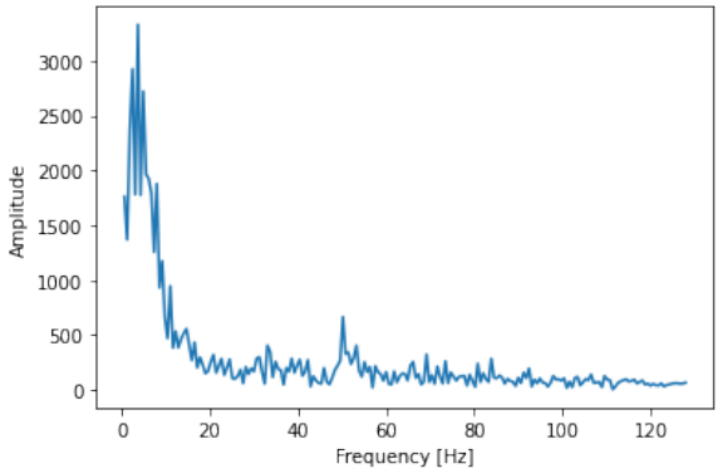

Figure 3.7: Plot of the signal in Figure 3.5 after application of the zero-padding and the FFT.

After applying the FFT algorithm to the windows, we applied dimensionality reduction in order to avoid overfitting through feature extraction. The obtained feature vectors used for classification, are composed of:

- The 10 frequencies with higher amplitude absolute value in the FFT for each of the four sensors (excluding the 0Hz frequency that is the highest amplitude in both 'distracted' and 'attentive' classes).
- The average of the window values for each brainwave for each of the four sensors.

Therefore, each feature vector consists of 60 features plus the distracted/attentive class.

3.2.3 Classification

For the classification of the feature vectors, we used different machine learning algorithms.

One of the most used classifiers to date for EEG-based brain-computer interfaces (BCIs) is the Support Vector Machine (SVM) [CST+00], especially for online and real-time BCIs [LBC+18]. It is a binary classifier and determines the hyperplane that provides maximum class separation. The idea of the method is to divide the feature

space, obtained by mapping incoming data into a higher dimensional space using a kernel function, into two parts using linear or non-linear decision boundaries. In the case of multiple classification problems, data is transformed into several problems with two classes.

Another accurate classifier is the Random Forest (RF) [Bre01]. In various problems and classification domains, RF algorithms have often been found among the most accurate classifiers, including problems with small training datasets. RF was also successfully used online for both event-related potential-based BCI and motor imagery BCI [LBC+18]. The idea behind this classifier is to randomly select a subset of the available features and to train a decision tree classifier on it, then repeat the process with many subsets of random features to generate many decision trees. The final decision is made by combining the results of all decision trees.

3.3 Experiments

Our technique has been evaluated using two cross-validation approaches. In the first approach, which we name *leave-one-person-out* cross-validation, 5 fold cross-validation was carried out, in which each fold corresponds to the data collected by a single volunteer. In the second approach, named *subject-specific* cross-validation, the data of each volunteer was taken into account separately, performing a sequential 5 fold cross-validation on each volunteer's dataset.

In order to evaluate the results of the classification, the components of the confusion matrix were first calculated, i.e.:

- True Positive (TP), which identifies the elements classified as belonging to the class 'attentive' and that belong to that class.
- False Positive (FP), which identifies the elements classified as belonging to the class 'attentive' and that belong to the class 'distracted'.
- True Negative (TN), which identifies elements classified as belonging to the class 'distracted' and that belong to that class.
- False Negative (FN), which identifies the elements classified as belonging to the class 'distracted' and that belong to the class 'attentive'.

The metrics used to evaluate the technique are the standard measures of accuracy, precision, recall, and F-score.

- Accuracy, which corresponds to the percentage of correct predictions in the classification of a test. It is equivalent to the ratio between the number of correct predictions and the total number of instances of the test set. However, when the classes are imbalanced, as in our case, this metric is not the most appropriate. Indeed, depending on the degree of imbalance, the majority class accuracy value can overcome the accuracy value of the minority class.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

- Precision corresponds to the number of TP divided by the total number of elements labeled as belonging to the actual class (the sum of TP and FP). Therefore, it indicates how many elements that are classified as part of the positive class actually belong to that class.

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

- Recall corresponds to the number of TP divided by the total number of elements that actually belong to the class (the sum of TP and FN) and indicates how effective the classifier is in identifying the positive label.

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

- F-score is the harmonic mean of precision and recall. This metric is useful to find a balance between precision and recall. It assumes optimal values close to 1.

$$F\text{-}score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{3.4}$$

The results obtained by applying the first cross-validation approach of the models are reported in Table 3.1.

The results obtained by applying the second cross-validation approach are shown in Table 3.2 for the SVM classifier, and in Table 3.3 for RF classifier.

Table 3.1: Results of the experiment using leave one person out cross-validation

	Accuracy	**Recall**	**Precision**	**F-score**
SVM	0.91	0.99	0.88	0.93
RF	0.95	0.99	0.94	0.96

Table 3.2: Results of the experiment using the SVM classifier and subject-specific cross-validation.

	Accuracy	**Recall**	**Precision**	**F-score**
Dataset volunteer 1	0.88	0.69	0.98	0.80
Dataset volunteer 2	0.92	1	0.91	0.95
Dataset volunteer 3	0.86	1	0.83	0.90
Dataset volunteer 4	0.92	1	0.91	0.95
Dataset volunteer 5	0.93	1	0.92	0.95
Average	0.90	0.94	0.91	0.91

Table 3.1 shows that the RF classifier achieves slightly better results than the SVM in solving the problem we are addressing. Indeed, the accuracy, precision, and the F-score of the RF (respectively 95%, 94%, and 96%) are larger than those achieved by SVM. This consideration can also be done by comparing RF and SVM results obtained using subject-specific cross-validation (Table 3.2 and Table 3.3).

As already mentioned in Section 3.1, there are several ways to assess whether a person can be considered distracted during the labeling task. The first way is to check whether the person has entered the wrong label, and we refer to this distraction as 'error'. The second way is to determine whether the person has exceeded a certain time threshold (that, for the sake of this work, we set to 1,988 seconds), and we refer to this distraction as 'over time'. Then, we consider three cases. A first case happens when both kinds of distractions are encountered; i.e., the labeler enters a wrong label also exceeding the time threshold. We refer to this case as 'error over time'. We name 'error no over time' the case in which the labeler enters a wrong label without

Table 3.3: Results of the experiment using the RF classifier and subject-specific cross-validation.

	Accuracy	**Recall**	**Precision**	**F-score**
Dataset volunteer 1	0.95	0.90	0.97	0.93
Dataset volunteer 2	0.95	0.98	0.95	0.96
Dataset volunteer 3	0.89	0.99	0.86	0.92
Dataset volunteer 4	0.96	0.98	0.96	0.96
Dataset volunteer 5	0.84	1	0.99	0.99
Average	0.92	0.97	0.95	0.95

exceeding the time threshold. Finally, we name 'no error over time' the case in which the labeler enters the correct label but exceeds the time threshold. After the division of the distracted class into the three cases, 70 images were labeled as 'error over time', 120 as 'error not over time', and 1090 as 'no error over time'.

Therefore, we decided to analyze our results concerning these distraction cases. The leave-one-person-out cross-validation approach results are shown in Table 3.4. The subject-specific cross-validation approach results are shown in Table 3.5 for the SVM classifier and in Table 3.6 for the RF classifier.

Table 3.4: Achieved recall using leave-one-person-out cross-validation

	error no over time	**error over time**	**no error over time**
SVM	0.00	0.62	0.75
RF	0.01	0.70	0.96

Comparing the results obtained in the two cross-validation approaches, we can observe a substantial difference. The cases of distraction are better identified by subject-specific cross-validation. In particular, in the subject-specific approach, the 'error no over time' distraction type obtains a recall of 0.92 using RF , compared to a recall of 0.01 achieved using leave-one-person-out cross-validation.

Table 3.5: Achieved recall using the SVM classifier and subject-specific cross-validation.

	error no over time	**error over time**	**no error over time**
Dataset volunteer 1	0.88	0.87	0.90
Dataset volunteer 2	0.88	0.84	0.88
Dataset volunteer 3	0.80	1.00	0.90
Dataset volunteer 4	0.93	1.00	0.99
Dataset volunteer 5	0.85	0.89	0.95
Average	0.87	0.92	0.92

Table 3.6: Achieved recall using the RF classifier and subject-specific cross-validation.

	error no over time	**error over time**	**no error over time**
Dataset volunteer 1	0.90	0.94	0.91
Dataset volunteer 2	0.94	0.67	0.96
Dataset volunteer 3	0.89	1.00	0.93
Dataset volunteer 4	0.93	1.00	0.99
Dataset volunteer 5	0.94	0.89	0.99
Average	0.92	0.90	0.96

These results indicate that, using a leave-one-person-out cross-validation approach, our method cannot recognize the 'error no over time' distraction type, but only the other two cases of distraction. We observe that the other two cases of distraction are straightforward to recognize, by simply considering the time taken by the labeler to annotate the image. Hence, the recognition of 'error no over time' distraction type is the actual objective of our work. The fact that the 'error no over time' distraction type can be recognized with good recall only using the subject-specific cross-validation approach indicates that our technique must be trained on the specific labeler. On the contrary, the model trained with other labelers' data is not effective for the objective that we are pursuing in this work.

Chapter 4

Using Portable EEG Sensors to Evaluate Human Attention

The purpose of the work presented in this Chapter is to assess a person's attention state through the analysis of EEG signals. The proposed system can be useful in different areas, such as learning, rehabilitation, and psychology, or even for monitoring workers who perform activities in which alertness is needed, such as drivers or surgeons. Recently, the availability of low-cost EEG sensors, that are more accessible to the end user, has increased. Therefore, we decided to compare the effectiveness of a low-cost sensor and a more expensive sensor in assessing a person's attention.

In Section 4.1 we illustrate the datasets used and collected with the two sensors, we explain how features are extracted, and how the classifier is trained and tested. In Section 4.2 we illustrate our results.

4.1 Material and methods

In our work, we have considered two datasets containing brainwave data on which we have applied the same feature extraction and classification techniques. The first dataset, named 'Image-labeling dataset', was acquired using an off-the-shelf portable EEG headset with 4 channels. The second dataset, named 'Epoc', was acquired using a more sophisticated EEG device with data acquired from 7 channels. We experimented the performance of machine learning algorithms in distinguishing attentive, distracted, and drowsed states of the individual based on EEG signal processing. In our experiments, only the preprocessing phase of EEG data diverges.

Indeed, the data of the Epoc dataset are raw, so it was necessary to employ Fast Fourier transform algorithms to extract Delta, Theta, Alpha and Beta brainwaves.

4.1.1 Image-labeling dataset

The first dataset was collected during the course of the work presented in Chapter 3 in which the objective was to evaluate, based on the EEG signal, the performance of annotators in labeling a series of images. A detailed description of the dataset can be found in Section 3.1.

As previously explained, for the EEG data collection, we used the Muse (Figure 2.2) which is a low-cost sensor consisting of 4 electrodes that can collect information on brain activity with a 256Hz sampling frequency in a non-invasive way. The Muse electrodes gather signals from channels TP9, AF7, AF8, and TP10. These electrodes are named and positioned according to the International System 10-20.

We used the mobile application Mind Monitor (Figure 3.4) along with the Muse sensor for receiving the EEG signals. Among the different data it returns, those used in this work are:

- The date and time of the recording.
- Brainwaves Delta, Theta, Alpha, Beta for each sensor.

The brainwave values are absolute band powers, based on the logarithm of the spectral power density (PSD) of the EEG data for each channel. These values are calculated internally by the Mind Monitor application with a data rate of 10Hz. The extracted brainwaves and their frequency bands are Delta 1-4Hz, Theta 4-8Hz, Alpha 7.5-13Hz, Beta 13-30Hz. Delta waves are related to deep sleep, unconsciousness, anesthesia, and lack of oxygen; Theta waves activity occurs when a person experiences emotional pressure, unconsciousness, or deep physical relaxation; Alpha waves are instead visible when an individual is in a state of consciousness, stillness, or rest, whereas when one is thinking, blinking or otherwise stimulated, this wave type disappears (alpha block); and finally Beta waves are evident when a person thinks or receives sensory stimulation.

The participant's task was to label indoor images that appeared randomly in a data annotation interface by selecting one of the eight buttons with the correct label (Figure 3.2). The task took 30 minutes to complete. At first, the application shows

an image. As soon as the labeler presses a button or does not press any button for 7 seconds, the application displays the next image.

The class assignment is different from that made in Chapter 3, in fact, the "attentive" and "distracted" classes were assigned to the first 10 and last 10 minutes of the recording, respectively, and thus are not assigned depending on whether the image label given by the labeller is correct or not, or whether the label is given or not.

4.1.2 Epoc dataset

The second dataset was taken from the work of [AKM19]. For the EEG data collection, the authors used the Emotiv Epoc+ kit EEG headset (Figure 2.1), which is a portable and non-invasive device consisting of 14 electrodes. The data are collected with a sampling rate of 128 Hz. The device was modified to allow electrode placement on the frontal and parietal areas of the scalp. Among the available channels, only O1, O2, P7, P8, AF4, F3, and F7, named and positioned according to the International System 10-20, were used in the presented work, since the other ones gave no insightful information for attention monitoring, or were affected by an excessive level of noise.

Since brainwaves data in the dataset are raw, we preprocessed the data by applying the fast Fourier transform [Nus81] to obtain Delta (0.5-4 Hz), Theta (4-8 Hz), Alpha (8-14 Hz), and Beta (14-30 Hz) brainwaves.

Data were collected from 5 volunteers who took part in 7 experiments in different days. The first 2 experiments were used to give the participant an understanding of the task, while the last 5 experiments were included in the dataset we used. The experiments were conducted between 7 p.m. and 9 p.m. to facilitate the third phase of the experiment involving a state of drowsiness. The participant's task consisted of controlling a train using the Microsoft Train simulator program, through simple keyboard commands, for a minimum duration of 30 minutes.

At the end of the task, the recording is divided into 10-minute fragments. Each one is related to a particular mental state:

- 'Attentive': The subject is focused on controlling the train, even though most of the task did not involve active intervention in the travel by the participants. First fragment.

- 'Distracted': The subject does not fall asleep but is distracted, and stops paying attention to the computer screen. Second fragment.
- 'Drowsed': The subject has no control over the train and keeps his eyes closed. Third fragment.

4.1.3 Feature extraction

The various brainwaves signals were divided into 10-second long non-overlapping sliding windows. For each window, we calculated the following 7 features:

- mean,
- median,
- variance,
- standard deviation,
- maximum,
- minimum,
- difference between maximum and minimum values.

These features are computed for each value of brainwaves Delta, Theta, Alpha, Beta, for each channel. Hence, we use 112 features for the Image-labeling dataset (4 channels), and 196 features for the Epoc dataset (7 channels), plus the class label.

4.1.4 Classification of human attention level

Feature vectors are used to train and test a Random Forest (RF) classifier [Bre01]. In various problems and classification domains, including problems with small training datasets, RF have often been found among the most accurate classifiers. RF and random trees were also successfully used for run-time brain-computer interface applications [LBC+18, MM20]. The RF randomly selects a subset of the available features to train a decision tree classifier on it; then it repeats the process with other subsets of random features to generate many decision trees. The final decision is made by combining the results of all decision trees using an ensemble approach.

4.2 Experimental evaluation

Our technique has been evaluated using two cross-validation approaches. In the first approach, named subject-specific cross-validation, the data of each volunteer was taken into account separately, performing a sequential 5 fold cross-validation on each volunteer's dataset. In the second approach, which we name leave-one-person-out cross-validation, k fold cross-validation was carried out, in which each fold corresponds to the data collected by a single volunteer.

The results obtained by applying the first cross-validation approach to the Image-labeling dataset are reported in Table 4.1 and the results obtained by applying the second cross-validation approach to the same dataset are shown in Table 4.2.

Table 4.1 displays very different results among the subjects, ranging from an Accuracy of 62% to 100%, probably due to the headband that is sensitive to movement and not easy to place in the correct position. The second approach obtained an average Accuracy of 61% (Table 4.2), probably due to inter-subject variability of acquired EEG data. Overall, the subject-specific approach achieves better results (80% average Accuracy) than the leave-one-person-out approach.

Table 4.3 and Table 4.4 report the results of the first approach applied to the Epoc dataset to solve the attentive/distracted and attentive/drowsed classification problems, respectively. Finally, Table 4.5 and Table 4.6 show the results obtained by applying the second approach to the Epoc dataset to solve the same problems.

We can make similar considerations to those made previously comparing the results of Tables 4.1 and Table 4.2, although in this case, the gap between the results obtained with the application of the two approaches is less evident. In particular, in the subject-specific approach, we have an overall Accuracy of 72% (Table 4.3) and 86% (Table 4.4), compared to an accuracy of 68% (Table 4.5) and 80% (Table 4.6) obtained using leave-one-person-out cross-validation.

Considering the Image-labeling dataset, we can observe that the average accuracy of distinguishing attentive and distracted states is 80% when we use a subject-specific cross validation approach; i.e., when the classifier is trained on the data of the same individual used for testing. Unfortunately, when we use a leave-one-person-out cross validation approach, the accuracy drops to 61%, which is a rather weak result for a binary classification problem. With the latter approach, we use more extensive training data, but those data are acquired from different people than the individual

Dataset	**Accuracy**	**Confusion Matrix**		
Tester 1	68%	Attentive	Distracted	
		41	19	Attentive
		19	41	Distracted
Tester 2	62%	Attentive	Distracted	
		47	13	Attentive
		32	28	Distracted
Tester 3	86%	Attentive	Distracted	
		49	11	Attentive
		5	55	Distracted
Tester 4	84%	Attentive	Distracted	
		50	10	Attentive
		9	51	Distracted
Tester 5	82%	Attentive	Distracted	
		48	12	Attentive
		10	50	Distracted
Tester 6	100%	Attentive	Distracted	
		60	0	Attentive
		0	60	Distracted
Overall	80%	Attentive	Distracted	
		295	65	Attentive
		75	285	Distracted

Table 4.1: Image-labeling dataset. Subject-specific cross-validation.

used for testing.

With the Epoc dataset, we achieved similar results. Indeed, the average accuracy of distinguishing attentive and distracted states is 72% when we use a subject-specific cross validation approach. With the same approach, the average accuracy of distinguishing attentive and drowsed states is 86%. The recognition achieved with the latter problem is higher, probably because drowsiness is easier to distinguish from attentiveness with respect to distraction. Also with this dataset, using a leave-one-person-out cross validation approach determines a considerable drop of accuracy; i.e., 68% accuracy in distinguishing attentive from distracted states, and 80% accuracy in distinguishing attentive from drowsed states.

These results indicate that our method achieves relatively high accuracy only when the system is trained with data acquired from the final user of the system. Training the system with data acquired from different persons determines a relevant drop in accuracy. This fact undermines the practical utility of this technique for some

Accuracy	**Confusion Matrix**		
61%	Attentive	Distracted	
	207	153	Attentive
	130	230	Distracted

Table 4.2: Image-labeling dataset. Leave-one-person-out cross-validation

Dataset	**Accuracy**	**Confusion Matrix**		
Tester 1	57%	Attentive	Distracted	
		175	125	Attentive
		131	169	Distracted
Tester 2	71%	Attentive	Distracted	
		234	66	Attentive
		107	193	Distracted
Tester 3	77%	Attentive	Distracted	
		229	71	Attentive
		59	231	Distracted
Tester 4	78%	Attentive	Distracted	
		196	44	Attentive
		62	178	Distracted
Tester 5	76%	Attentive	Distracted	
		174	66	Attentive
		50	190	Distracted
Overall	72%	Attentive	Distracted	
		1008	372	Attentive
		409	961	Distracted

Table 4.3: Epoc dataset. Subject-specific cross-validation. Attentive/distracted classification.

applications, since the system would require an initial training phase by the user which may be time-expensive and uncomfortable. This problem may be addressed by using transfer learning methods explicitly proposed for EEG data [WYH+21].

Another worth noting finding of our experiment is that the more sophisticated device used for the Epoc dataset achieves essentially the same accuracy of the simpler device used for the Image-labeling dataset. This result indicates that even an off-the-shelf device may be effective to support some attention-aware applications.

Dataset	**Accuracy**	**Confusion Matrix**		
Tester 1	75%	Attentive	Drowsed	
		204	96	Attentive
		52	248	Drowsed
Tester 2	87%	Attentive	Drowsed	
		281	19	Attentive
		58	242	Drowsed
Tester 3	89%	Attentive	Drowsed	
		289	11	Attentive
		53	247	Drowsed
Tester 4	92%	Attentive	Drowsed	
		223	17	Attentive
		20	220	Drowsed
Tester 5	89%	Attentive	Drowsed	
		210	30	Attentive
		21	219	Drowsed
Overall	86%	Attentive	Drowsed	
		1207	173	Attentive
		204	1176	Drowsed

Table 4.4: Epoc dataset. Subject-specific cross-validation. Attentive/drowsed classification.

Accuracy	**Confusion Matrix**		
68%	Attentive	Distracted	
	940	440	Attentive
	429	951	Distracted

Table 4.5: Epoc dataset. Leave-one-person-out cross-validation. Attentive/distracted classification.

Accuracy	**Confusion Matrix**		
80%	Attentive	Drowsed	
	1094	286	Attentive
	240	1140	Drowsed

Table 4.6: Epoc dataset. Leave-one-person-out cross-validation. Attentive/drowsed classification.

Chapter 5

GNN for HD EMG-based Movement Intention Recognition

In the work presented in this Chapter, we proposed the use of a GNN architecture for amputee movement intention recognition based on HD-EMG electrodes data. For building the graph, we considered 32 ms sliding windows, since shorter window sizes can be processed faster, leading to shorter controller delays and, consequently, better experience for the user. We experimented our methods with a real-world dataset acquired from 20 participants wearing on the forearm two HD-EMG electrodes with 64 channels each to recognize 65 gestures.

Section 5.1 reports some information about the public HD-EMG dataset we used in our work. Section 5.2 explains our methodology. Section 5.3 reports our experimental evaluation.

5.1 HD-EMG dataset

In order to evaluate our method, we conducted extensive experiments with a recently released dataset [MOS+21]. Data were recorded at the forearm level from 20 able-bodied participants (14 men and 6 women) aged between 25 and 57 (mean age 35). Each participant performed five repetitions of 65 gestures, reported in Table 5.1, with a rest period of 5 seconds between each repetition.

Table 5.1: The 65 classified gestures with their respective labels, their description, and their complexity expressed in DoF.

Class	**Gesture**	**DoF**
1	Little finger: bend	1
2	Little finger: stretch	
3	Ring finger: bend	
4	Ring finger: stretch	
5	Middle finger: bend	
6	Middle finger: stretch	
7	Index finger: bend	
8	Index finger: stretch	
9	Thumb: down	
10	Thumb: up	
11	Thumb: left	
12	Thumb: right	
13	Wrist: bend	
14	Wrist: stretch	
15	Wrist: rotate anti-clockwise	
16	Wrist: rotate clockwise	
17	Little finger: bend+Ring finger: bend	2
18	Little finger: bend+Thumb: down	
19	Little finger: bend+Thumb: left	
20	Little finger: bend+Thumb: right	
21	Little finger: bend+Wrist: bend	
22	Little finger: bend+Wrist: stretch	
23	Little finger: bend+Wrist: rotate anti-clockwise	
24	Little finger: bend+Wrist: rotate clockwise	
25	Ring finger: bend+Middle finger: bend	
26	Ring finger: bend+Thumb: down	
27	Ring finger: bend+Thumb: left	
28	Ring finger: bend+Thumb: right	
29	Ring finger: bend+Wrist: bend	

30	Ring finger: bend+Wrist: stretch	2
31	Ring finger: bend+Wrist: rotate anti-clockwise	
32	Ring finger: bend+Wrist: rotate clockwise	
33	Middle finger: bend+Index finger: bend	
34	Middle finger: bend+Thumb: down	
35	Middle finger: bend+Thumb: left	
36	Middle finger: bend+Thumb: right	
37	Middle finger: bend+Wrist: bend	
38	Middle finger: bend+Wrist: stretch	
39	Middle finger: bend+Wrist: rotate anti-clockwise	
40	Middle finger: bend+Wrist: rotate clockwise	
41	Index finger: bend+Thumb: down	
42	Index finger: bend+Thumb: left	
43	Index finger: bend+Thumb: right	
44	Index finger: bend+Wrist: bend	
45	Index finger: bend+Wrist: stretch	
46	Index finger: bend+Wrist: rotate anti-clockwise	
47	Index finger: bend+Wrist: rotate clockwise	
48	Thumb: down+Thumb: left	
49	Thumb: down+Thumb: right	
50	Thumb: down+Wrist: bend	
51	Thumb: down+Wrist: stretch	
52	Thumb: down+Wrist: rotate anti-clockwise	
53	Thumb: down+Wrist: rotate clockwise	
54	Wrist: bend+Wrist: rotate anti-clockwise	
55	Wrist: bend+Wrist: rotate clockwise	
56	Wrist: stretch+Wrist: rotate anti-clockwise	
57	Wrist: stretch+Wrist: rotate clockwise	
58	Extend all fingers (without thumb)	>= 3
59	All fingers: bend (without thumb)	
60	Palmar grasp	
61	Wrist: rotate anti-clockwise with the Palmar grasp	
62	Pointing (index: stretch, all other: bend)	

63	3-digit pinch	>=3
64	3-digit pinch with Wrist: anti-clockwise rotation	
65	Key grasp with Wrist: anti-clockwise rotation	

The 65 gestures include:

- individual fingers flexions and extensions;
- thumb flexions, extensions, abductions and adductions;
- wrist flexions, extensions, pronations and supinations;
- some combinations of the above movements;
- some of the most common synergistic multi-joint hand movements.

For EMG data collection, the authors used two HD-sEMG electrodes, each consisting of 64 channels arranged in an 8×8 matrix with an inter-electrode spacing of 10 mm (ELSCH064NM3, OT Bioelettronica, Turin, Italy). The electrodes were placed approximately 3 cm from the elbow (elbow to closest electrode corner) and 2 cm from the ulna (edge of the ulna to edge of the electrode). The sensing device is shown in Figure 5.1

The EMG signals were sampled at 2048 Hz. A hardware high-pass filter at 10 Hz and a low-pass filter at 900 Hz were used during recordings. To reduce the noise in the EMG signal consecutive channels were subtracted during the registration. Due to the orientation of the electrodes relative to the underlying muscles, the subtraction of the EMG signals was done along with the muscle i.e. ch1 signal was calculated as the difference between EMG signals at electrode contacts 2 and 1, ch2 as the difference between signals at contacts 3 and 2, and so on.

5.2 Methodology

In this Section, we explain how, based on the structure of the HD-EMG electrodes used, the graph was created and we give details on the EMG-GNN architecture.

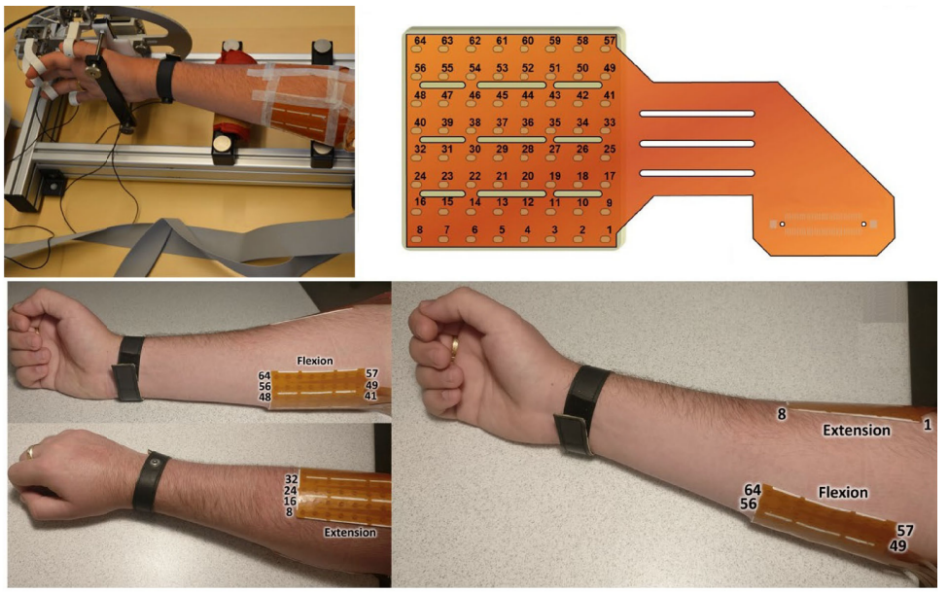

Figure 5.1: In [MOS+21] the HD-sEMG electrodes, visible in more detail in Figure 2.5, are positioned approximately 3 cm from the elbow and 2 cm from the ulna.

5.2.1 Graph-based modeling of HD-EMG data

A graph consists of a set of nodes and edges connecting them. In our graph-based model, each channel used to collect EMG data represents a node of the graph. To connect nodes by means of edges, several strategies have been proposed in related works [DKAW+21], including:

1. each pair of nodes is connected by an edge without a feature,
2. each pair of nodes is connected by an edge whose feature is the Pearson correlation coefficient between the feature vectors of the two nodes,
3. only nodes closer than a heuristic distance are connected, or
4. nodes are connected through the use of k-nearest neighbours (k-NNG).

For the sake of this work, we decided to use the 3rd approach where the distance is 15mm. The resulting topology is shown in Figure 5.2. As shown in the Figure, in order to simultaneously consider the data acquired from the two electrodes, we added edges from the nodes of the first electrode's last row to the nearest nodes of the second electrode's first row.

Graph nodes possess characteristics so each of them is associated with a feature vector corresponding to a sliding window containing a time sequence of 65 samples acquired from the respective channel. The sliding windows size generally used to decode EMG signals is greater than 100 ms [KN21]. However, a key aspect when working with prostheses is the response of the system, which can be improved by decreasing the size of the sliding windows used. For this reason, we divided the signals from the different channels using non-overlapping sliding windows of 65 samples, corresponding to 32 ms of recording. Before division into sliding windows, the EMG signals were standardized, which implies scaling the distribution of values so that the mean of the observed values is 0 and the standard deviation is 1.

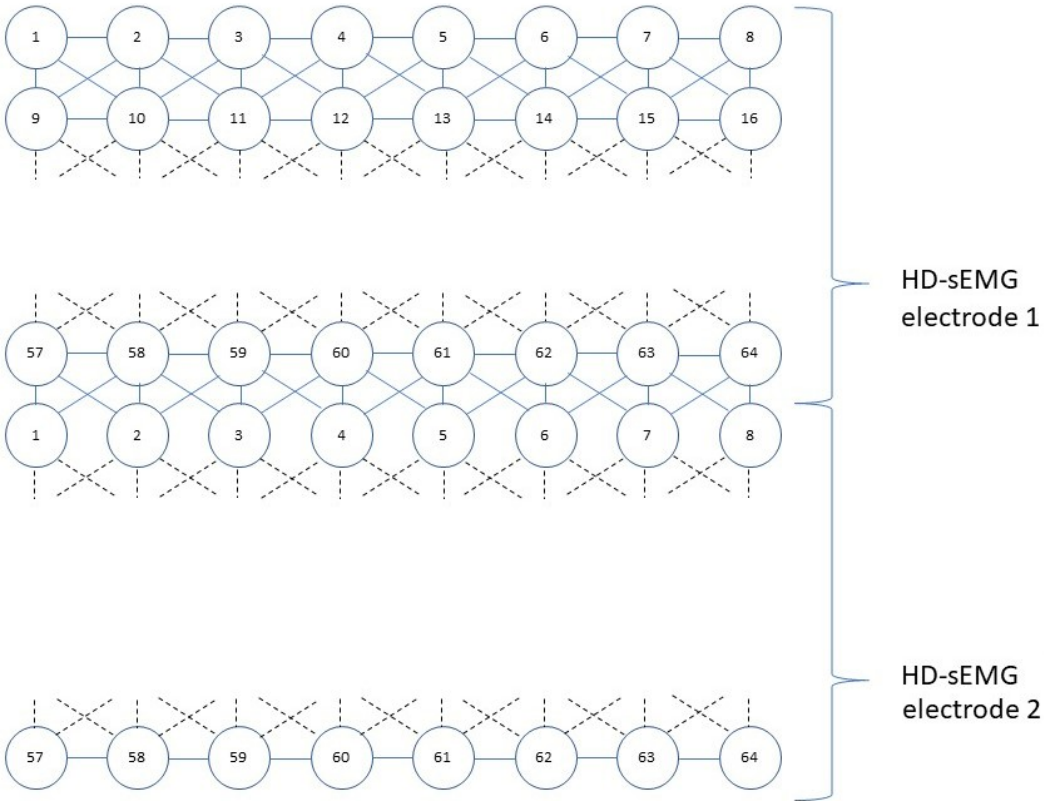

Figure 5.2: The graph consists of 128 nodes and 884 edges. Each node corresponds to a channel. Each node is connected, with not oriented edges, to nodes distant less than 15 mm. For each node, the signal acquired from the respective channel is split using sliding windows. Each sliding window corresponds to the feature vector of the node. The nodes in the first row of the second electrode are connected to the nearest nodes in the last row of the first electrode. The structure of its channels is analogous to the organization of pixels in an image.

5.2.2 EMG-GNN Structure

The GNN structure is analogous to the one proposed in [DKAW+21] for EEG signal classification (Figure 5.3).

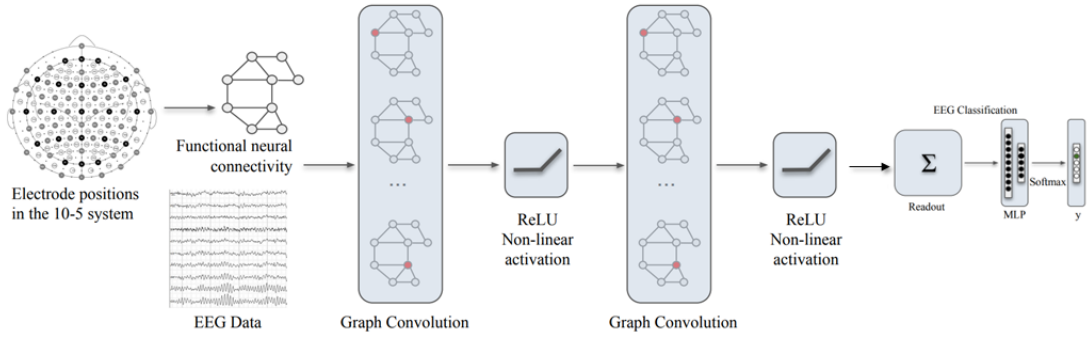

Figure 5.3: Schema of the EEG-GNN structure presented in [DKAW+21].

The structure of our EMG-GNN is shown in Figure 5.4, and it consists of:

- graph convolutional layers and ReLU non-linearity applied to the signals mapped onto the graph structure to embed each node by performing multiple rounds of message passing;
- a READOUT function to learn the representation vector of the entire graph through the aggregation of the node representations from the final graph convolutional layer;
- a multi-layer perceptron (MLP) to classify the graph representation vector;
- a Dropout layer before the final layer to avoid overfitting;
- a Linear activation at the output layer.

SAGEConv implements the GraphSAGE operator proposed in [HYL17]. GraphSAGE is a general inductive framework that leverages node feature information to efficiently generate node embeddings for previously unseen data. This framework is designed for large graphs with a high number of nodes. GraphSAGE learns a function that generates embeddings by sampling and aggregating the local neighborhood features of a node, unlike most existing approaches that require all nodes in the graph to be considered during embedding training.

The number of layers and neurons was decided after several experiments.

We used the Adam Optimizer with a starting Learning Rate (LR) of 0.001. We also used ReduceLROnPlateau, which reduces the LR when a metric has stopped improving for a "patience" number of epochs. In our case, we monitor the Validation Loss, and if its value does not decrease for 10 epochs, the learning rate is reduced by 0.1.

We applied Cross-Entropy Loss and monitored the Validation Loss to decide when to stop training. The Early Stopping allows us to speed up learning and to avoid overfitting. If the Validation Loss value does not decrease for 30 epochs, model training is stopped; otherwise, the model is trained until the 100th epoch is executed.

We batched the graphs, setting the size to 32, before putting them into the GNN to ensure full GPU utilisation.

The GNN we employed in this work was developed in Python programming language using PyTorch Geometric (PyG) which is a library built upon PyTorch to easily write and train GNNs for a wide range of applications related to structured data [PyG].

5.3 Experiments

We decided to consider the different subjects' datasets separately during the trials. Therefore, before each trial, we shuffled the graphs of a single subject's dataset, and then divided them into 60% training set, 20% validation set, and 20% test set.

The metric used to evaluate the technique is the error rate, which corresponds to the percentage of incorrect predictions in the classification of a test.

$$Error\,rate = \frac{FP + FN}{TP + TN + FP + FN} \tag{5.1}$$

Table 5.2 shows, for each gesture identified by a number in the first column, the standard deviation and the overall classification error rate (%) obtained by considering all the results achieved during the execution of the different trials. As mentioned above, only one subject's dataset was used during each trial. A usable system should achieve error rate levels less than 10% [SE11]. Then, we highlighted in red the gesture classification error rates higher than this percentage value, i.e.,

20 gestures out of 65.

The most difficult gestures to recognize are:

- 10 Thumb: up
- 19 Little finger: bend + Thumb: left
- 24 Little finger: bend + Wrist: rotate clockwise
- 26 Ring finger: bend + Thumb: down
- 27 Ring finger: bend + Thumb: left
- 29 Ring finger: bend + Wrist: bend
- 30 Ring finger: bend + Wrist: stretch
- 31 Ring finger: bend + Wrist: rotate anti-clockwise
- 32 Ring finger: bend + Wrist: rotate clockwise
- 34 Middle finger: bend + Thumb: down
- 35 Middle finger: bend + Thumb: left
- 38 Middle finger: bend + Wrist: stretch
- 43 Index finger: bend + Thumb: right
- 52 Thumb: down + Wrist: rotate anti-clockwise
- 53 Thumb: down + Wrist: rotate clockwise
- 56 Wrist: stretch + Wrist: rotate anti-clockwise
- 57 Wrist: stretch + Wrist: rotate clockwise
- 59 All fingers: bend (without thumb)
- 64 3-digit pinch with Wrist: anti-clockwise rotation
- 65 Key grasp with Wrist: anti-clockwise rotation

The above list shows that the EMG-GNN has difficulty detecting mainly complex gestures with 2 or more degrees of freedom (DoF).

In the last row of Table 5.2, we reported the standard deviation and the overall classification error rate (%) obtained by considering all the results achieved during the execution of the different trials, without taking into account the subdivision into gestures.

These results, obtained with a baseline GNN implementation, are well aligned with the state of the art, and support the importance of further investigation of our approach.

Table 5.2: Standard deviation and overall error rate (%). Classification error rates greater than 10% are highlighted in red. A usable system should achieve error levels below 10%.

Gesture	**Standard deviation**	**Overall error rate (%)**
1	4,14	4,32
2	7,13	6,16
3	6,11	6,58
4	5,84	6,16
5	4,43	4,89
6	6,79	6,74
7	5,88	6,84
8	5,49	4,26
9	7,78	8,63
10	6,66	11,68
11	6,87	9,42
12	7,1	9,11
13	3,09	3,37
14	5,38	4,37
15	3,39	4,47
16	4,54	4,63
17	7,11	6,63
18	7,49	8,95

19	8,89	10,21
20	5,35	8,16
21	6,12	6,89
22	6,6	8,74
23	6,66	7,74
24	7,19	10,26
25	6,64	7,42
26	9,52	14,42
27	10,16	14,58
28	5,06	8,47
29	8,89	12,26
30	9,15	12,84
31	10,1	12,37
32	8,98	12,21
33	3,78	6,74
34	5,32	10,32
35	6,94	11,21
36	5,67	8,74
37	7,88	8,63
38	10,18	12,37
39	5,77	7,95
40	7,52	8,84
41	4,94	6,84
42	7,2	9,58
43	8,65	11,16
44	7,03	8,05
45	7,3	9,42
46	6,96	8,84
47	8,39	9,26
48	8,46	8,26
49	5,9	6,58
50	5,27	7,79

51	5,9	7,26
52	9,67	11,58
53	10,05	11,53
54	8,06	8,53
55	7,12	8,47
56	11,27	12,89
57	6,84	10,21
58	4,31	4,74
59	11,9	11,42
60	6,13	8,32
61	6,46	8,63
62	5,46	7,05
63	5,57	7,74
64	10,83	14,58
65	8,93	11,63
Overall error rate (%)	**4,92**	**8,75**

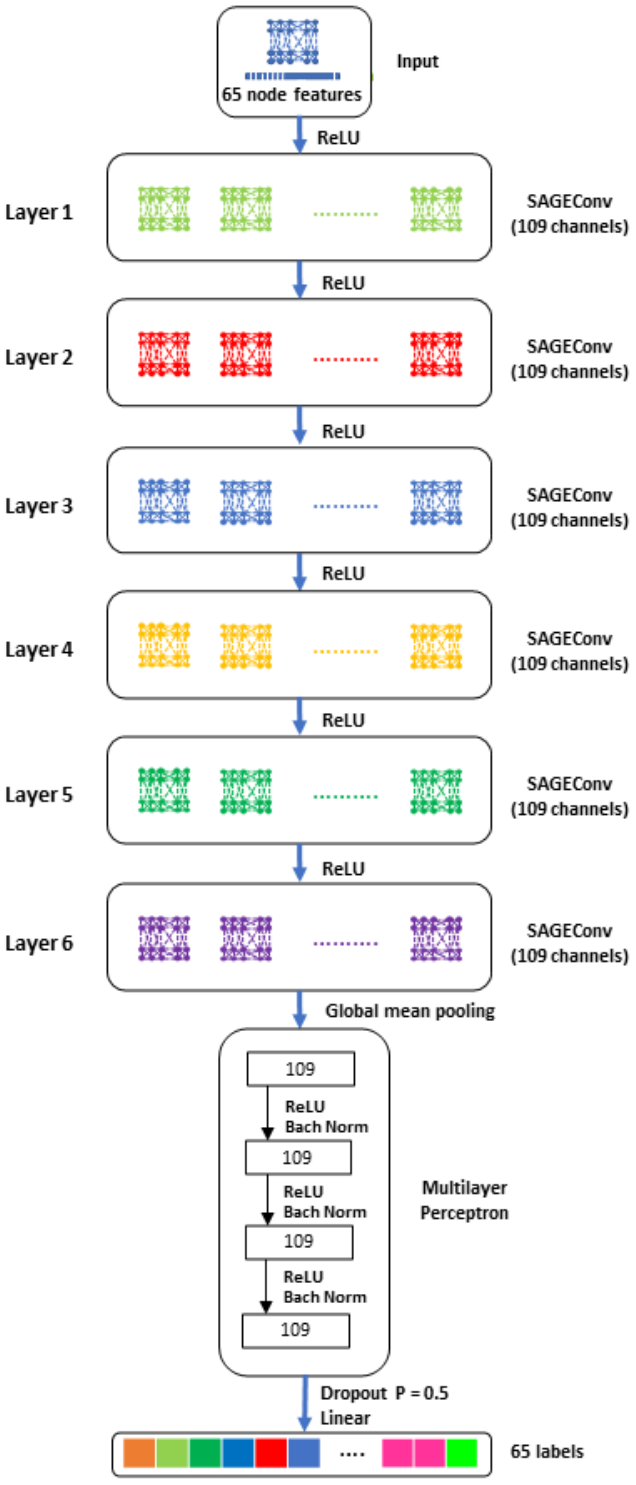

Figure 5.4: Schema of the EMG-GNN structure

Chapter 6

Recognition of Cooking Activities Through Air Quality Sensor Data

In this Chapter, we presented a food journaling system for frail people living alone, in which food preparation activities in the home are detected by exploiting data from air quality sensors. In terms of non-invasiveness and privacy, the technique has clear advantages over solutions based on wearable sensors or cameras.

How the air quality sensor data were acquired and how features have been engineered in order to feed a deep neural network is covered within Section 6.1. The experiments we have carried out, along with the results we have obtained, are presented in Section 6.2. Section 6.3 illustrates the prototype we have developed for the proposed use case.

6.1 Acquisition and processing of air quality sensor data

In this Section, we explain how we acquire and process air quality sensor data in order to recognize the preparation of food. The diagram in Figure 6.1 shows our framework for data acquisition and processing. An indoor air quality monitor deployed in the kitchen is in charge of providing a stream of real-time sensor data to a DATA CLEANING module. That module performs data preprocessing to eliminate possible errors in sensor readings. Then, the data is passed to a FEATURE EXTRACTION module, that builds feature vectors based on statistics computed on the

current and past data. The feature vector is passed to the ONLINE RECOGNITION module, which uses an Artificial Neural Networks classifier to detect whether the user at home is cooking or not. The Neural Network is trained in advance using a labeled training set of sensor data acquired during cooking and non-cooking activities. Finally, the prediction (either *cooking* or *not cooking*) is communicated to a robot, who is in charge of interacting with the user in order to interactively collect his/her food diary.

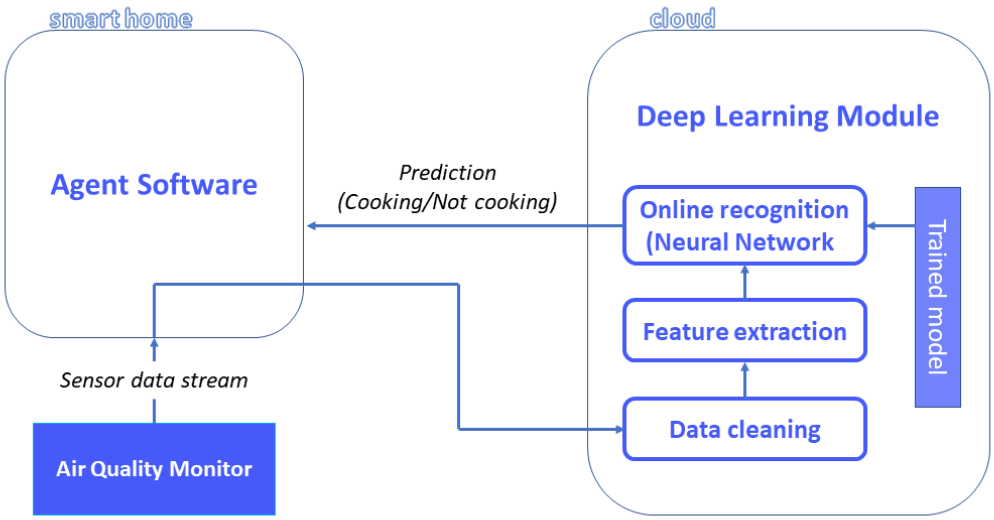

Figure 6.1: System that automatically recognizes cooking activities using an air quality sensor.

6.1.1 Sensor data acquisition

As shown in our experiments, reported in Section 6.2, the act of preparing food determines relevant changes in the air quality of the cooking area. In particular, the use of a gas cooker determines an immediate increase in the carbon dioxide (CO_2) level in the kitchen. The preparation of certain kinds of food generates fumes containing different levels of volatile organic compounds (VOCs) [CWL+16]; the increase of VOC levels is particularly evident when certain foods are prepared, such as meat and fish. Similarly, cooking certain kinds of food determines the emission of particulate matter (PM); i.e., microscopic matter suspended in the air. The

concentration and size of particulate matter are determined both by the cooking style (roasting, frying...) and by the used ingredients [ADSH13]. Natural gas stoves also emit other gases, such as NO_2, in the kitchen. Moreover, when cooking takes place, the environmental parameters of the kitchen are affected both in terms of temperature and humidity values.

Nowadays, indoor air quality monitors are becoming popular, due to their low cost and increased attention of people to the healthiness of indoor air. Our intuition is that it is possible to exploit off-the-shelf indoor air quality sensors in order to recognize food preparation activities by applying machine learning techniques to the sensor data stream. The advantage of this solution with respect to other ones based on cameras or environmental sensors is that the indoor air quality sensor is unobtrusive and requires negligible installation effort. Moreover, it is obviously more privacy-conscious than solutions based on microphones and cameras.

At the time of writing, different indoor air quality monitors are available on the market. These devices mainly differ from one another with respect to the kind of monitored parameters, detection frequency, form factor, network interfaces, presence of open APIs, and cost. For recognizing food preparation activities, we target a device having the following characteristics:

- it is able to monitor at least the following parameters: temperature, humidity, carbon dioxide, volatile organic compounds, particulate matter;
- it provides a detection frequency of at least one sensor reading per minute;
- for ease of installation, it provides a wireless network interface and electrical connection to avoid battery exhaustion;
- it provides open APIs for acquiring the sensor data in real-time.

Among several indoor air quality meters currently available on the market, we chose the uHoo device introduced in Section 2.3, which provides all the desired characteristics mentioned above. In addition to the mentioned parameters, the uHoo device also measures nitrogen dioxide, carbon monoxide, ozone, and air pressure. It provides sensor readings with a frequency of one minute, and the data can be downloaded either in batch from a smartphone app, or in real-time thanks to open APIs.

6.1.2 Data cleaning

In general, sensor data are affected by a relevant level of noise. Hence, before being used, the raw sensor readings must be preprocessed to reduce the noise, which could negatively affect the accuracy of inferred data. However, several air quality monitors, including the ones we use in this work, perform an internal preprocessing of the raw data before sending them to the user or application. Preprocessing usually consists in smoothing the values of consecutive readings, in order to correct values affected by high level of noise. Since smoothing is already performed internally by the air quality monitor, in this work we perform a limited data cleaning, which consists of disregarding those portions of consecutive data where more than 50% of values are missing due to network errors or power failures.

6.1.3 Feature engineering

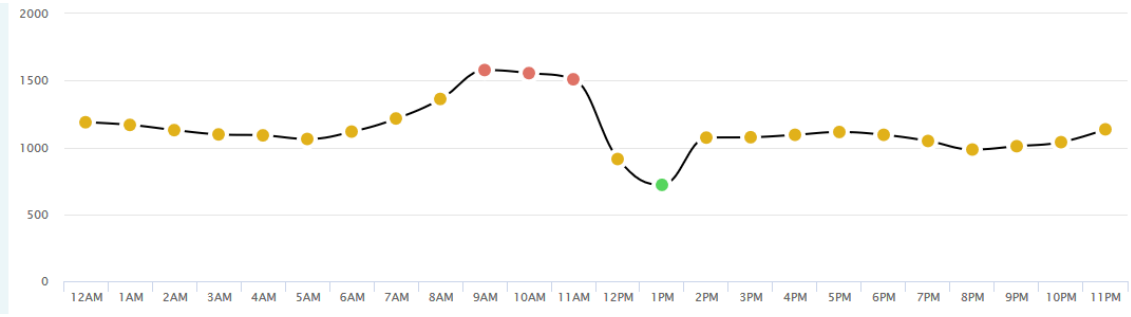

Figure 6.2: Hourly trend of carbon dioxide in the kitchen in a day. Each point represents the average carbon dioxide value in the kitchen during a given hour. The points can take on different colors: green represents comfortable values for human life; red represents uncomfortable values; yellow represents intermediate values.

In order to reliably recognize food preparation activities, it is necessary to provide the machine learning algorithm with features useful to discriminate between cooking and non-cooking activities. For this reason, we have carefully analyzed the trend of air quality data when cooking was performed or not.

Figure 6.2 shows a screenshot of our air quality Web dashboard. The plot depicts the trend of carbon dioxide hourly average during a day. On that day, breakfast and lunch were prepared at around 7:30 a.m. and at around 1:30 p.m., respectively. From the plot, it is easy to observe that the absolute value of carbon dioxide is not sufficient to reliably distinguish cooking from non-cooking activities. Indeed, during

all that day, the value of CO_2 was relatively stable, with a value slightly above 1,000 ppm. The value of CO_2 started to increase in the morning at 7:00, when people went to the kitchen and initiated preparing breakfast. The increase in carbon dioxide levels was due both to the breathing of people in the kitchen and the usage of a natural gas stove. The CO_2 value reached a local maximum at 9-10 a.m., and kept stable until 12:30 p.m., when a user opened the window to ventilate the kitchen. Soon after, a person started the preparation of lunch, and this activity determined an increase in carbon dioxide, whose value reached 1,000 ppm and remained stable for the rest of the day.

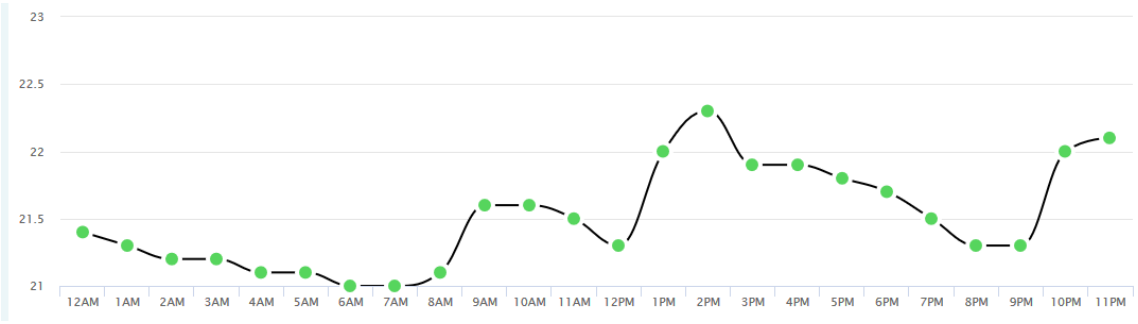

Figure 6.3: Hourly trend of temperature in the kitchen in a day. Each point represents the average temperature value in the kitchen during a given hour. Green points represent comfortable values for human life.

From the analysis of Figure 6.2, it emerges that, in order to distinguish cooking from non-cooking activities, it is important to analyze the trend of CO_2 levels, not only its absolute value. A similar point holds for the other parameters, such as the temperature, whose plot on the same day is reported in Figure 6.3. For this reason, we engineered features taking into account not only the absolute values or averages, but also the difference between the current value and the past values. In particular, we build features considering the differences between the most recent value and the one in the previous 5, 10, 15, 20, and 25 minutes. We also use statistical features considering the average, minimum, and maximum value in the last 5 minutes, as well as the standard deviation of those values. Using these features, which are built using only the current values and past values, it is possible to recognize the current activity online; hence, we name this feature engineering modality *online feature extraction*.

However, especially to reliably determine the end of the cooking activity, it

would be useful to observe the sensor data even after the end of the cooking activity. Indeed, the end of a cooking activity is often characterized by a drop of certain parameters, such as temperature, CO_2, and particulate matter; hence, the difference between those values during and after the cooking activity might generate characteristic spikes that are easy to recognize. Obviously, the use of features computed considering succeeding values determines a delay in the recognition process. Hence, for those applications having real-time requirements, such as the one addressed in this work, we only use features considering current and past values. For all the other applications, we also build features considering succeeding values in a temporal sliding window of 25 minutes. For instance, in order to build the feature vector referring to the activity executed at 12:00, we need to wait until 12:25, since the feature vector is built based on data acquired from 11:35 to 12:25. We name this feature engineering modality *delayed feature extraction*. In our experiments, reported in Section 6.2, we evaluate both modalities.

It is worth to note that a single parameter is not sufficient to reliably recognize cooking activities. In general, increasing levels of carbon dioxide indicates the preparation of food using a gas stove; however, there may be some false positive when several people are in the kitchen, especially if the window is closed and the kitchen is small or poorly ventilated. Relying on CO_2 only, false negatives may happen when a meal is prepared without using a gas stove. For instance, in the day concerning Figures 6.2 and 6.3, a dinner was prepared at around 9 p.m. using an electric oven. We can observe that the increase of CO_2 in that period of time was very limited, and due only to the sporadic presence of one person in the kitchen. The usage of the oven was clearly captured by the increase of the temperature. However, temperature alone is not a reliable parameter for food preparation, since it is strongly influenced by climatic factors and other external conditions. For this reason, we build our feature vectors considering six sensor data parameters: temperature, humidity, carbon dioxide, volatile organic compounds, particulate matter, and nitrogen dioxide. We disregarded carbon monoxide, ozone, and air pressure, because we experimentally found that they were not reliable indicators of food cooking.

Finally, the time of the day is another important indicator of food preparation, since cooking is normally carried out at specific times. We compute the current time of the day as the number of minutes that passed from midnight.

6.1.4 A deep neural network for food preparation

We have built a deep neural network for the classification task. The type of deep neural network chosen is a Multilayer Perceptron (MLP). Figure 6.4 shows its architecture. In particular, our MLP is composed by four layers: one input layer, two hidden layers, and one output layer. The layers are fully connected (dense). The units per layer have been selected considering the number of features. Let nF be the number of features in input. The input layer has $nF/2$ number of units, the first hidden layer has exactly nF number of units, the second hidden layer has $2nF$ number of units. The output layer has only one unit, since we are performing a binary classification. We have used the Leaky Rectified Linear Units function (LeakyReLU) as activation function, with negative slope coefficient set at 0.2. Instead, for the output layer, we chose as activation the Sigmoid function, since we need a binary output value.

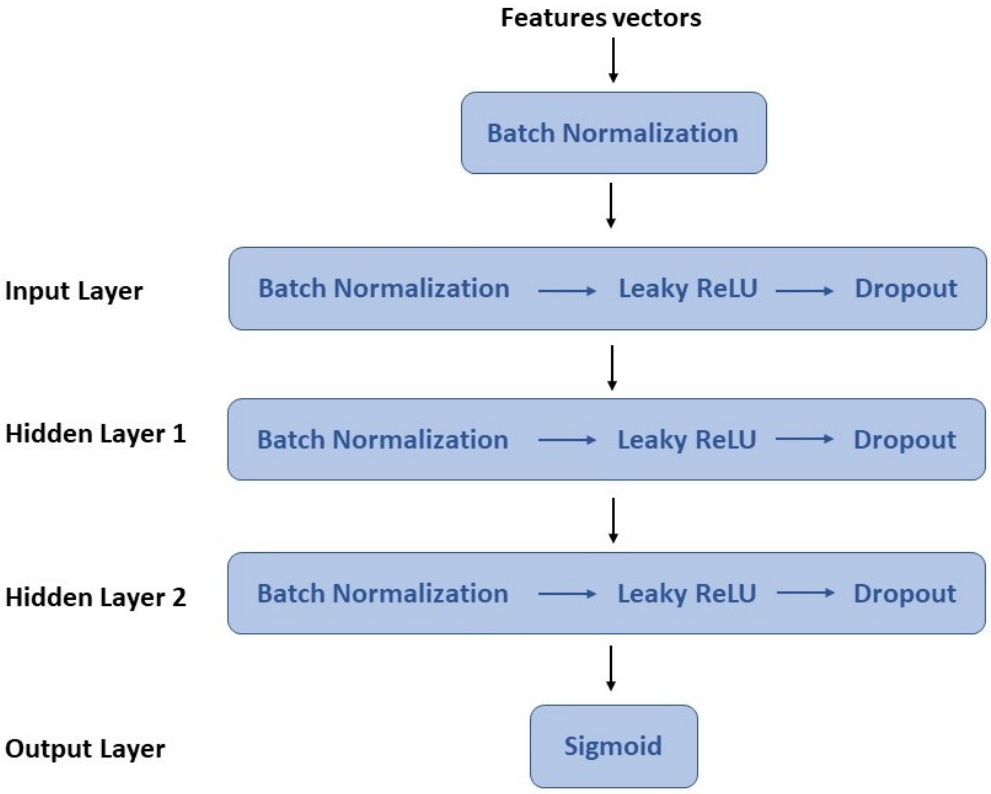

Figure 6.4: Architecture of the proposed deep neural network.

To prevent over-fitting we have added Dropout layers after every LeakyReLU layer. The fraction of the input units to drop has been set at 0.5. To speed up learning and to increase the stability of the neural network, we have also added Batch Normalization layers: one before the input layer and the other before every LeakyReLU layer. Batch Normalization layers have allowed us using a low learning rate (set at 0.0001) with Adam chosen as optimizer. As loss function, we used the binary cross-entropy function.

The deep neural network we have employed in this paper has been developed in Python programming language using the Keras framework [ker] and Scikit-Learn library [sci]. The used environment has been Google Colaboratory [col].

The collected sensor data, the related annotations provided by humans, and the code related to the deep neural network we have developed can be freely downloaded from a GitHub repository[1].

6.2 Experimental evaluation

In this Section, we report the results of experiments carried out with an extensive set of real-world data.

6.2.1 Dataset

The dataset is composed of 350,551 data readings taken at each minute during more than 8 months in total from volunteers living in 8 different homes. The participants self-annotated the start and end time of cooking activities on a printed form, also specifying the kind of food that they cooked. At the end of data acquisition, the annotations were digitized by researchers using a custom program. The researchers actively interacted with the participants to clarify the meaning of those annotations that were ambiguous or unclear. The dataset was acquired in real-world environments and in naturalistic conditions; we did not rely on multiple annotators and we could not evaluate inter-rater reliability. As a consequence, even though the participants took annotation with care, the self-annotations inevitably may contain missing or wrong labels [TBW+18]. Data have been collected in homes having different characteristics, and at different periods of the year, to guarantee diversity and

[1] https://github.com/FG2511/MLP_ForFoodPreparation

to ensure that the data represented real situations and conditions. In particular, six homes were situated in a city area by the sea with a Mediterranean climate, one in a big continental city, and one home was situated in a mountain area with an alpine climate. Climate influences temperature and humidity, and the area (city vs countryside) may influence air pollutants, particulate matter, and volatile organic compounds. In five homes, data was affected by the presence of people in the kitchen after the completion of the cooking activity, while in the other homes the meals were consumed in a different room. The season influences the frequency of other activities, such as opening the window or turning on a heating system, that may affect ambient and air parameters. The participants' age ranged from 23 to 71 years, with 10 females and 8 males. The volunteers were recruited among the families and mates of the authors, in order to include different kinds of inhabitants. Specifically, homes included six different typologies of inhabitants: middle-aged single inhabitants, couples, families with children, groups of roommate students, a senior living alone, and a senior living with a middle-aged person. The participants did not receive any compensation for taking part in the study. All volunteers were informed about the procedure used for data acquisition, the kind of data that would be acquired, the frequency of acquisition, and the kind of sensitive information that could be extracted from the acquired data. We explained that the data could be released in anonymous form to third parties for research purposes. In particular, we explained that we would not release any micro-data to third parties. Instead, we would release only aggregated macro-data; i.e., statistical feature vectors to be used for classification. Released data would not include neither explicit identifiers, nor quasi-identifier information. We also explained the potential impact of the research for supporting several kinds of medical conditions. The volunteers gave written informed consent to their participation to the experiments. Each data record contains:

1. date,
2. time,
3. temperature (in $^{\circ}C$),
4. relative humidity (in percentage %),
5. PM2.5 (Fine Particulate Matter in $\mu g/m^3$),

6. TVOC (Total Volatile Organic Compound in *ppb*),
7. CO_2 (Carbon Dioxide in *ppm*),
8. CO (Carbon Monoxide in *ppm*),
9. air pressure (in hPa),
10. O_3 (Ozone in *ppb*),
11. NO_2 (Nitrogen Dioxide in *ppb*),
12. current activity,
13. type of cooked food (e.g., rice, salad).

The *current activity* attribute can take two values: 1 if the user is cooking a meal, 0 otherwise. The number of data records with activity set to 1 is 16,323, while the remaining 334,228 are set to 0, meaning that "cooking" and "not cooking" classes are strongly unbalanced. As an example, Table 6.1 indicates a few records of the dataset with the information above.

Timestamp	**Temp.**	**Hum.**	**PM2.5**	**TVOC**	**CO_2**	**CO**	**Pres.**	**O_3**	**NO_2**
2018-11-24 13:13	27.7	60.14	5.54	66.0	442.0	0.0	1012.91	9.15	28.60
2018-11-24 13:14	27.7	60.21	4.56	67.0	461.0	0.0	1012.92	9.14	28.76
2018-11-24 13:15	27.7	59.84	8.37	67.0	465.0	0.0	1012.89	9.35	32.29
2018-11-24 13:16	27.6	58.96	6.19	67.0	467.0	0.0	1012.91	9.57	36.25

Table 6.1: Few examples of dataset records. Each record is annotated with the current activity (1 if the user is cooking a meal, 0 otherwise), and the list of cooked food.

The data are subject to many variables: the kind of person who cooked most in the house (3 men and 5 women, ages ranging from 20 to 72), the sensor distance from domestic appliances used for cooking (ranging from 5 cm to 1.5 m), the presence of air conditioning, pellet stove or windows in the kitchen, and the house structure (separate kitchen from the dining room or open-space). The data were acquired in different seasons; this feature strongly affects some parameters such as temperature and humidity.

Eight volunteers were given one air quality monitor to place in their homes, and for one month each they collected sensor data by writing down specific information

each time they cooked: date, start and end time of cooking, cooked foods, domestic appliances used for cooking, and presence of an open window. The composition of home inhabitants was disparate, and included couples, students sharing a house, elderly living alone, and families with children.

6.2.2 Experimental setup

In order to optimize the deep neural network, it has been necessary to perform preliminary experiments to fine-tune the model parameters: activation function, optimizer, learning rate, batch size, dropout rate value. The optimized module that we devised is the one described in Section 6.1.4. As the classes in the dataset are strongly unbalanced (as explained in Section 6.2.1), the class weights have been set up before the model generation.

Hence, we carried out several experiments, using two different types of validation.

1. Initially, the model has been evaluated using a one-shot split of the dataset. The model took 80% of the dataset as training set, the following 10% as validation set, and the remaining 10% for testing. The number of epochs has been decided using the Early Stopping function, which stops the evaluation when the loss function starts to increase. The patience parameter (i.e., the number of epochs with no improvement after which training stops) has been set to 2.

2. With the second type of validation, the model has been tested using a 10 fold cross-validation. Specifically, we have used the Scikit-Learn KFold function to split the dataset. The split has been done maintaining the temporal order of the dataset by setting the shuffle parameter to False. This peculiarity is important, since shuffling the instances can introduce bias. Indeed, two instances that are contiguous in the dataset (i.e., two set of data measured at one-minute distance) are very similar: if an instance goes to the training set and the following goes to the test set, we have a bias.

We evaluate our model using two modalities. The first modality is named "minute-by minute", and considers each prediction, referring to a one-minute data, in isolation. The second modality is named "cooking instance" recognition, and refers to the recognition of whole instances of cooking, where a cooking instance is a continuous interval of time during which the cooking activity took place.

For minute-by-minute modality, a correct recognition of *cooking* at minute m is counted as a true positive (TP). A false positive (FP) happens when the network predicts *cooking* at m, but cooking was not taking place at that minute. A false negative (FN) is counted if cooking was occurring at m, but the reasoner wrongly predicts "not cooking" for that minute. Finally, a true negative happens when the reasoner correctly predicts that cooking is not taking place at m.

For cooking instance modality, we consider each segment of contiguous minute-by-minute predictions of "cooking" starting at minute m and ending at minute n as the prediction of a single instance of cooking. Then, we count a TP if an actual instance of cooking has an interSection with a predicted cooking instance. If it has no interSection, then we count it as a FN. A FP occurs when (i) an actual instance of "not cooking" contains a predicted cooking instance, or (ii) a predicted instance of "cooking" contains an actual instance of "not cooking". A TN occurs when a predicted instance of "not cooking" does not contain an actual "cooking" instance.

The metrics used to evaluate the model are: accuracy, precision, recall, and F_1 score

The results obtained have been improved with a post-processing step. The post-processing has been developed using a simple sliding windows algorithm. We have used More Itertools [mor] library to implement sliding windows. The length of the windows has been set to 35 minutes. In each window, we look at the class of the central element (e.g., class 1), then we count the elements belonging to the same class. If they are less than a certain threshold (set at the half of the window plus one), the central element is set to the other class (e.g., class 0). In the other case, the class of the central element remains the same. The purpose of this step has been to remove small clusters of outliers and to merge close clusters of the same class.

6.2.3 Results

In the following, we present the experimental results. In all experiments, we applied 10-fold cross-validation. Since air quality data values change relatively slowly with time, we have built the folds sequentially, in order to avoid the risk of having consecutive data instances (which may be very similar among them, if not even identical) appearing in the training and test set, which could bias our results.

At first, we evaluated the performance of classification using different state-of-the-art machine learning algorithms. In these experiments, we used the Weka

toolkit [HFH+09] for machine learning. Results obtained with minute-by-minute modality are shown in Table 6.2. Results show that the classification problem we are addressing is particularly challenging. Overall, among the evaluated classifiers, the one achieving highest accuracy was Random forest (95.95% accuracy). However, it is well known that, especially when classes are unbalanced, accuracy alone is not an adequate metric to evaluate the effectiveness of classification. Indeed, that classifier obtains good precision (70.57%), but low recall (22.62%), meaning that the predictions of "cooking" were quite reliable, but most cooking instances were actually not recognized. The classifier obtaining the highest F_1 score (35.63%) was Bayes networks, that (contrary to Random forest) exhibited good recall (69.88%), but low precision (23.91%). The Naive Bayes and Logistic regression classifiers obtained lower recognition rates than the former algorithms. The kNN classifier achieved a very good balance between precision (28.15%) and recall (28.85%); however, its overall recognition performance was low (F_1 score = 28.5%). The Support Vector Machines classifier obtained one the highest scores for recall (64.65%), but the lowest score for precision (20.71%), reaching an F_1 score of 31.37%.

	R.F.	**B.N.**	**N.B.**	**L.R.**	**kNN**	**SVM**
TN	332208	297465	308529	332056	321735	293351
FP	1539	36282	25218	1691	12012	40396
TP	3691	11403	7519	2665	4707	10549
FN	12626	4914	8798	13652	11610	5768
Accuracy	**95.95%**	88.23%	90.28%	95.62%	93.25%	86.81%
Specificity	**99.54%**	89.13%	92.44%	99.49%	96.40%	87.90%
Recall	22.62%	**69.88%**	46.08%	16.33%	28.85%	64.65%
Precision	**70.57%**	23.91%	22.97%	61.18%	28.15%	20.71%
F1-Score	34.26%	**35.63%**	30.66%	25.78%	28.50%	31.37%

Table 6.2: Results with minute by minute modality, online recognition. Classifiers: Random forest (denoted as R.F., max depth = 10, iterations = 10), Bayes networks (B.N., using K2 hill climbing search algorithm), Naive Bayes (N.B.), k nearest neighbor (kNN, using $k = 1$), Logistic regression (L.R.), Support Vector Machines (SVM, using polynomial kernel and class balancing).

Then, we performed classification using our deep learning model. Table 6.3 summarizes the results of online recognition obtained with minute-by-minute modality. Before post processing, despite the overall accuracy obtained being 87.95%, the overall F_1 score was slightly higher than 36%. Hence, the overall accuracy was

	Original Predictions	**After Post-Processing**
TN	296219	301258
FP	37978	32939
TP	12055	11985
FN	4269	4339
Accuracy	87.95%	**89.36%**
Specificity	88.64%	**90.14%**
Recall	**73.85%**	73.42%
Precision	24.09%	**26.68%**
F_1 **score**	36.33%	**39.14%**

Table 6.3: Deep neural network. Results with minute by minute modality, online recognition.

comparable to the one obtained by the Bayesian network classifier, which was the one achieving the highest F_1 score in our pool of classifiers. However, our neural network obtained higher recall (73.85% vs 69.88%) and essentially the same precision (24.09% vs 23.91%). Moreover, the size of the dataset was relatively small for training a deep neural network. We expect that the results of our deep neural network may significantly increase using additional training data. For these reasons, we decided to use the deep neural network in the rest of the experiments. The relatively low recognition rates that we achieved may be probably due to the fact that the dataset is strongly imbalanced, since time of cooking covers less than 5% of the dataset. For this reason, it was hard for the neural network to identify the few "cooking" activities within the vast majority of "not cooking" instances. Moreover, the dataset was acquired in several different real-world conditions. In particular, our neural network achieved good recall, but low precision. Results were slightly improved by post-processing, reaching an F_1 score close to 40%. By inspecting the results, we observed that post-processing improved the precision by around 3% without negatively impacting recall. We repeated the same experiments with delayed recognition. We recall from Section 6.1.3 that in this modality the recognition of the current activity is delayed by 25 minutes in order to consider the succeeding trend of air quality values. With minute-by-minute recognition, we observed that delayed recognition achieved essentially the same accuracy of online recognition, as shown in Table 6.4. We performed a statistical study in order to understand whether the difference in the results obtained with online vs offline recognition was statistically

significant. For this reason, we applied the well-known measures of Φ coefficient and χ^2 test [Gui41] to the output of the classifiers using the two recognition methods. We recall that the Φ value of two binary variables having identical distribution tends to 1, while the p value of χ^2 test tends to 0. In our case, we obtained a Φ value of 0.90, and the p value of the χ^2 test of 2.2e-16. Hence, we can conclude that the two techniques produced results that are statistically very similar for minute-by-minute classification.

	Original Predictions	**After Post-Processing**
TN	293027	297008
FP	41140	37159
TP	12687	12461
FN	3637	3863
Accuracy	87,22%	**88,30%**
Specificity	87,69%	**88,88%**
Recall	**77,72%**	76,34%
Precision	23,57%	**25,11%**
F_1 **score**	36,17%	**37,79%**

Table 6.4: Deep neural network. Results with minute by minute modality, delayed recognition.

However, for our application, it is important to identify whole instances of cooking activities, not the single minutes during which the activity takes place. In cooking instance modality, our online recognition method achieved better results than in minute-by-minute modality. Results can be found in Table 6.5. In particular, before post-processing, the technique achieved an F_1 score slightly lower than 46%. This modality significantly increased both precision (from 24.09% to 32.21%) and recall (from 73.85% to 78.77%). Results were further improved by post processing, reaching an overall F_1 score close to 60%. In particular, post-processing provided more balance between precision and recall values. Note that, after post-processing, the total number of predicted instances was strongly reduced, and this fact had obviously an impact on the overall numbers of TN, FP, TP, and FN. The reduction of the total number of predicted instances was due to the fact that our post-processing algorithm merged multiple predicted cooking instances that were temporally close. The reader is referred to Section 6.2.2 for the definition of cooking instance modality. In cooking instance modality, accuracy improved using delayed

recognition (Table 6.6), achieving an F_1 score larger than 62%.

	Original Predictions	**After Post-Processing**
TN	4506	727
FP	1109	482
TP	527	491
FN	142	178
Accuracy	**80.09%**	64.86%
Specificity	**80.25%**	60.13%
Recall	**78.77%**	73.39%
Precision	32.21%	**50.46%**
F_1 **score**	45.73%	**59.81%**

Table 6.5: Deep neural network. Results with cooking instance modality, online recognition

	Original Predictions	**After Post-Processing**
TN	3988	698
FP	957	454
TP	549	505
FN	120	164
Accuracy	**80,82%**	66,06%
Specificity	**80,65%**	60,59%
Recall	**82,06%**	75,49%
Precision	36,45%	**52,66%**
F_1 **score**	50,48%	**62,04%**

Table 6.6: Deep neural network. Results with cooking instance modality, delayed recognition

6.2.4 Discussion

Overall, despite we carefully designed the deep neural network, the achieved results are not fully satisfactory. This fact may be explained in several ways.

- First of all, while the dataset includes both hot and cold meals, our system is suited to recognize only the former. Indeed, it fails to recognize the majority of cold meals. This is an intrinsic limitation of any recognition system based

on air quality data. In order to recognize cold meals, different kinds of sensors should be added to the system.

- Secondly, the dataset was acquired in disparate real-world conditions. Homes included single inhabitants, couples, families with children, and groups of roommate students. Of course, the age and number of inhabitants has an impact on the kind and quantity of cooked food, and consequently on the change in air quality conditions determined by cooking. The topology of the home also has an impact on air quality data. Indeed, if inhabitants consume the meal within the kitchen, their presence determines an increase of temperature and CO_2 levels even after cooking has ended. If the inhabitants consume the meal in a different room, the CO_2 and temperature levels in the kitchen decrease as soon as cooking is finished. In our dataset we had both cases, depending on the home. This aspect could be taken into account by selecting only the subset of the training data acquired in conditions that resemble those of the target environment.

- Thirdly, being manually annotated, the dataset labels have an inevitable level of noise, which may include wrong start and end time of cooking execution, or wrong labels.

Nonetheless, considering that each activity recognition system has a considerable error rate, our system based on air quality data can be coupled with other activity recognition tools in the home to increase the overall activity recognition rate. For instance, the accuracy of the system may be increased by coupling our air-quality based system with other sensors attached to kitchen furniture and instruments. Moreover, as explained in Section 6.3, the user's feedback resulting from the interaction with the robot is used to periodically re-train the neural network using additional training data. Hence, we expect the accuracy of the system to increase with time thanks to human-robot interaction. Even though we did not evaluate this aspect in our experiments, we also believe that the number of false positives may be significantly reduced thanks to the usage of computer vision APIs of the robot, as described in Section 6.3.2.

The average execution time of the neural network algorithm for recognizing an instance of data is 0.0317 milliseconds on a cloud computing infrastructure. Hence, our system is feasible for real-time applications as in the proposed use case.

6.3 Use case on a robotic platform

In this Section we are going to describe the use case we have set within the social robotics domain. A humanoid robot has been employed to interact with the user when the system recognizes that something is being cooked. In such a case, the robot asks the user what he/she is cooking. More in detail, Section 6.3.1 will include details of the robotic platform we have adopted whereas Section 6.3.2 will include the architecture of the use case we have designed.

6.3.1 Zora, the used humanoid robot

The Zora robot [zor] uses the same robotic infrastructure of Nao, an autonomous, programmable humanoid robot developed by Aldebaran Robotics, a French robotics company headquartered in Paris, which was acquired by SoftBank Group in 2015 and re-branded as SoftBank Robotics. With respect to Nao, Zora adds an extremely simple and intuitive user interface that allows any user to play loaded behaviors (apps, dances, and games targeting care, kids, STEM market), to give action commands to the robot to change posture and move each part of her body, to make her talk in eight possible languages, and to use the Composer to create simple robot behaviors, composing a sequence of actions in a visual environment where no programming knowledge is needed.

Like Nao, Zora is also completely programmable through the Choregraphe suite [cho], which allows users to:

- create and combine different robot behaviours using a visual approach making use of the Python programming language;
- develop animations by leveraging an intuitive and dedicated user interface;
- test the robot behaviours and animations on either the simulated robot or the real one;
- develop complex behaviours and human-robot interactions by leveraging calls to REST APIs of external services on the Internet.

In order to capture the voice of the user when he/she speaks, the robot is equipped with four microphones, two of them in the front of the head and two

Figure 6.5: An image of Zora, the employed humanoid robotic platform.

at the back. The robot can therefore record the human voice, which is contextually analyzed and transformed into text by a speech recognition module powered by Nuance [nua]. However, we are currently relying on cloud computing systems for speech recognition in order to improve the accuracy of the speech to text process. In fact, this allows us pre-processing the sound recorded by Zora and removing noise (e.g background noise, fan noise, etc.), which may compromise the conversion of human voice into written text. As such, the resulting audio file is sent to IBM Watson Speech to Text [wat] to perform speech recognition. Figure 6.5 shows an image of Zora.

6.3.2 Architecture of the use case

Figure 6.6 shows the architecture of the proposed use case. A Deep Learning module contains the annotated data and the trained deep learning model. That module exposes REST APIs to classify as *cooking* or *non cooking* a new collected record of sensor data. One more software agent, periodically, collects sensor data and calls the REST APIs of the Deep Learning module. If the new read data is classified as *cooking* then this is communicated to the robot via a socket communication. Before starting the interaction with the user, the robot checks if someone is actually in the kitchen. For such a purpose, it takes a picture of the environment, which

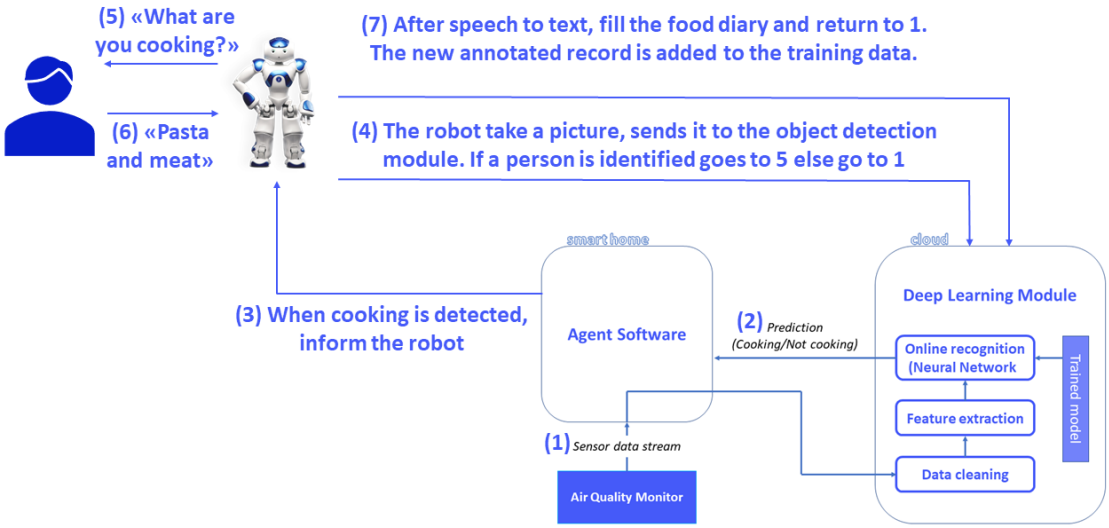

Figure 6.6: Overall architecture of our use case prototype implementation. Note that point (4) is optional: the user may enable or not the camera-based object detection of the robot. If (4) is disabled, action goes directly from point (3) to point (5).

is sent to an object detection module to identify potential persons. However, for the sake of privacy, the object detection task is optional, and the user may decide whether to activate it or not. More specifically, we have employed the TensorFlow Object Detection API [ten], which provides an open-source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models. TensorFlow Object Detection API can be used with different pre-trained models. More in detail, we have chosen a Single Shot MultiBox Detector (SSD) model with Mobilenet (ssd_mobilenet_v1_coco) which had been trained using the Microsoft COCO dataset[COC], consisting of 2.5M labeled instances in 328000 images, containing 91 object types. The ssd_mobilenet_v1_coco is reported to have mean average precision (mAP) of 21 on the COCO dataset. For further details on the SSD and the evaluation carried out on the COCO dataset please check the work of authors in [LAE+16]. The back-end of the object detection module has been embedded into a server-side application which exposes REST APIs that, given an input image, return the bounding box of each recognized object in the image along with a category and a confidence value. We considered valid only the objects that

were recognized with a confidence value equal or higher than 60%. The back-end is hosted within the Deep Learning Module.

When a cooking instance is recognized, the robot starts the interaction with the user. If camera-based recognition is enabled, the robot takes six pictures in the kitchen, each 60 degrees distant from the other. If the robot identifies one or more persons (class *person* of the COCO dataset) in the images, it asks what food the user is preparing. If camera-based recognition is disabled, the robot makes its question in any case. Once the user replies, the robot performs speech to text processing and sends the extracted food as well as the sensors data to the Deep Learning module which extends its training data with the new annotation and, periodically, retrains the overall model. Note that, in the current implementation of our system, the speech interaction to acquire food journaling data is over-simplistic, being based on a simple question-answer paradigm. Since most food journaling applications require detailed information about the kind and quantity of food, we will investigate a more sophisticated conversational agent for food journaling in future work. Voice-based identification methods will also be used to recognize the inhabitant, in case of multi-resident homes. If the classifier forecasts a *cooking* activity using the current air quality data and the user is not actually cooking (i.e. the user replies *nothing* to the robot question above) the classification is wrong and the new record is sent to the Deep Learning Module together with the *not cooking* label. If the object detection module does not identify any person in the kitchen, it overrides the classification of the Deep Learning Module thus reducing the false positives and improving the overall classification. Moreover, the new pair (sensor data, *not cooking*) is sent as further training element. Periodically and when enough new annotated data have been collected, the Deep Learning Module trains again the model.

We would like to point out that the robot has not been employed for the collection of the annotated data during the eight months from the annotators. During that period, only our air quality sensors were employed and their measurements were saved for the whole period. After the creation of our gold standard and the training of our model, we set up the whole architecture shown in Figure 6.6 for a preliminary test on real settings. Advanced methods to improve our use case, including techniques for optimized path planning of the mobile robot [SRR20], will be investigated in future work.

Time	Food
8:11	coffee
13:18	pasta and potatoes
17:09	chocolate
20:23	broccoli and steak
22:34	tea

Table 6.7: Five entries of the food diary filled during one day through the human robot interaction use case.

6.3.3 Preliminary results on human-robot interaction mechanism

After having collected all the information related to the air quality sensors, in one of the houses of the annotators we performed a preliminary technical validation of the human-robot interaction mechanism. In order to interact with the user, the robot employs a state of the art object detection classifier, text-to-speech and speech-to-text technologies, which are widely evaluated in the literature. We have already mentioned the used speech-to-text technologies. As far as the object detection is concerned, we have used the classifier based on the work of authors in [LQQ+18]. The object detection software (that was enabled in our use case) and the classification software (to identify a cooking instance out of the air sensor data) were run in a pc we brought in the house to perform the test together with the robot and the air sensors. The whole human-robot interaction architecture has been preliminarily tested for short time (one full day), and the only errors we noticed occurred because of the wrong prediction of the activity recognition module. As mentioned earlier in the paper, the human-robot interaction has been kept simple. To easily recognize the food spoken by the user, we first collected a list of food items online that were enriched by each of the annotators. Basically, we asked each of them to write the list of food items they have cooked or might cook in the future. After we removed the duplicates, we obtained a list of 86 food items that were organized in a two-levels hierarchical structure. The first layer contained general terms, whereas the second levels contained items that were associated to one food item of the first level. For example, *egg* is a general item, while *omelette* is a specific item related to *egg*. Therefore, when the machine learning module predicted a cooking activity (and the robot identified a person in the kitchen), the robot asked the user what

he/she was cooking. Out of the natural language expressions spoken by the user, after the robot performed speech-to-text, it was just a matter of recognizing terms we had in the vocabulary without performing any comprehension of the semantics involved in the natural language text. This process did not lead to any errors and all the spoken items have been correctly identified within the defined vocabulary. There was one researcher present in the morning during the first cooking activity and in the evening during the last cooking activity that monitored the human robot interaction after having trained the English speaking person living in the house and informed her on the behaviour of the robot. Some facts, comments and impressions that turned out from our preliminary experiment were the following:

- the object detection module correctly identified all the times when someone was in the kitchen;
- one out of five cooking activities was not recognized as a cooking activity by the classifier;
- two times the robot thought that there was a cooking activity (two false positives occurred of the Deep Learning module): that was fixed as soon as the user replied *nothing* to the robot question *What are you cooking?* and the correct pair (sensor data, no cooking activity) was sent to the Deep Learning module;
- we have developed everything (vocabulary, human robot interaction, etc.) in English and the user involved within our experiment was an English speaker;
- out of five cooking activities, there were not cases when the user mentioned a food not present within the dictionary we had prepared;
- the entries we filled in our food diary for the short experiment are depicted in Table 6.7;
- we asked what the impressions of the user interacting with the robot were and she was very curious and excited to talk it. She did not think the robot was intrusive and liked the simple human robot interaction we designed. She would even have liked if the robot could have entertained her with songs, music, radio, or simple interaction or question-answering capabilities provided, for example, by voice assistant tools today.

Chapter 7

Conclusions

In this thesis, we worked on different types of disorders that affect the mind and body by first creating ad-hoc solutions for specific disabilities and finally creating a versatile interactive system that can be useful to people with different types of disabilities.

In the first work, we introduced the use of EEG data mining to assess the performance of humans carrying out annotation tasks. Our method relies on a consumer EEG sensor and on supervised machine learning. We have collected a dataset from five volunteers. Initial results indicate that our approach is promising. This work can be improved in several directions. Our results indicate that the system is reliable when the training set is acquired from the specific individual for which it is used. However, the utility and scalability of the system should be improved by using training data from different individuals. To this aim, we will investigate the use of transfer learning methods specifically tailored to EEG data. We will also investigate cost-effective techniques to acquire training data from the specific subject. Finally, we will acquire a larger dataset to thoroughly evaluate our techniques.

In the first part of the thesis, we also conducted a study to understand the influence of the cost of the EEG device used in estimating the level of attention. We then tested the data collected through two sensors, a low-cost and a more expensive one, using the same techniques for feature extraction and classification. The results obtained are very similar and in both cases, the system obtains better values when the classifier is trained on the data of the same individual used for testing. Future work includes investigating different machine learning algorithms for the classification task, including deep learning methods, to improve the accuracy of the system,

and feature selection techniques to reduce overfitting.

In the third work, we have introduced a novel approach, using GNNs for processing HD EMG data to support the movement intention recognition of amputees. The use of GNNs allows the modeling of complex topological relations of the electrodes, which are not captured by traditional machine learning algorithms or by other deep learning architectures. An investigation of our method, including experiments with a real-world dataset, shows that the approach is promising. Future work includes a deeper investigation of the spatiotemporal characteristics of HD EMG data to refine the graph structure. We are also considering using explainable artificial intelligence methods to investigate the internal functioning of the deep learning model to fine-tune the network structure for reducing computational costs. Finally, we will experiment with our methods with additional datasets, and perform an experimental comparison with state-of-the-art techniques.

In the last presented work, we have laid the foundation of a novel method to support food journaling, addressed to frail people who live alone or with family members who are away from home for most of the day and cannot constantly take care of them. Our system relies on advanced air quality sensors for cooking recognition. We have shown the process of collecting and analyzing air quality sensor data to detect when the user is cooking in order to trigger the interaction with a digital agent to acquire food data. We have developed a deep neural network trained on a large dataset acquired over 8 months in disparate conditions by different people. An experimental evaluation has been carried out to assess the accuracy of the model on the given classification problem and the feasibility of the method for real-time applications. We have also developed an initial prototype considering a use case where a social robot interacts with the neural network and with the user. Our preliminary prototype is the first in its kind and shows several challenges we had to face and many more that we still need to address. However, we believe that the technologies to address these challenges are out there and our work provides a significant step in this direction. Several challenges to be addressed in future work remain open. First of all, we will investigate methods to increase the accuracy of our cooking recognition system. An obvious direction is to couple the air quality sensor with other sensors to recognize the preparation of cold meals. As explained, the temperature alone is not sufficient to reliably recognize cooking activities, because indoor temperature is influenced by external temperature. A similar point holds for

humidity and other factors. In order to mitigate the influence of external conditions, we could include additional data taken from online weather services. Since both the topology of the home and the characteristics of inhabitants (including their number and age distribution) affect the air quality conditions at cooking time, we will investigate techniques to couple our data-driven method with a knowledge-based one, to fine-tune recognition to home's and inhabitants' characteristics. Other domain knowledge, such as the expected duration of cooking activities, may be used to improve the recognition rates of our cooking recognition system, and this is a research direction we will pursue. Future work also includes the definition of an effective and engaging conversational interface for interactively filling the food diary, and voice-based identification methods to recognize the current inhabitant in the case of multi-resident homes. For such a purpose, one direction we are heading is to employ Google Assistant technology for the human-robot interaction exploiting the APIs and open-source tools (e.g. DialogFlow) that Google makes available to the community. We would like to extend the vocabulary we have defined according to Semantic Web best practices in order to have a more comprehensive ontology involving all the food items that might be cooked according to any international recipes. Finally, we plan to execute extensive tests and a much more comprehensive evaluation of the human-robot interaction approach based on our preliminary use case prototype implementation

Bibliography

[ADSH13] Karimatu L Abdullahi, Juana Maria Delgado-Saborit, and Roy M Harrison. Emissions and indoor concentrations of particulate matter and its specific chemical components from cooking: A review. *Atmospheric Environment*, 71:260–294, 2013.

[AIK+09] Md Rezwanul Ahsan, Muhammad I Ibrahimy, Othman O Khalifa, et al. Emg signal classification for human computer interaction: a review. *European Journal of Scientific Research*, 33(3):480–501, 2009.

[AIR+15] Amna Abdullah, Asma Ismael, Aisha Rashid, Ali Abou-ElNour, and Mohammed Tarique. Real time wireless health monitoring application using mobile devices. *International Journal of Computer Networks & Communications (IJCNC)*, 7(3):13–30, 2015.

[AK21] Chloe Agg and Samana Khimji. Perception of wellbeing in educational spaces. *Building Services Engineering Research and Technology*, 42(6):677–689, 2021.

[AKA20] Alessandra Angelucci, David Kuller, and Andrea Aliverti. A home telemedicine system for continuous respiratory monitoring. *IEEE Journal of Biomedical and Health Informatics*, 25(4):1247–1256, 2020.

[AKM19] Cigdem Inan Aci, Murat Kaya, and Yuriy Mishchenko. Distinguishing mental attention states of humans via an eeg-based passive bci using machine learning methods. *Expert Syst. Appl.*, 134:153–166, 2019.

[AP15] Abdulla Ali and Sadasivan Puthusserypady. A 3d learning playground for potential attention training in adhd: A brain computer interface approach. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 67–70. IEEE, 2015.

[Art17] Ron Artstein. Inter-annotator agreement. In *Handbook of linguistic annotation*, pages 297–313. Springer, 2017.

[ASEE18] Alaa Eddin Alchalabi, Shervin Shirmohammadi, Amer Nour Eddin, and Mohamed Elsharnouby. Focus: Detecting adhd patients by an eeg-based serious game. *IEEE Transactions on Instrumentation and Measurement*, 67(7):1512–1520, 2018.

[ASLT05] Oliver Amft, Mathias Stäger, Paul Lukowicz, and Gerhard Tröster. Analysis of chewing sounds for dietary monitoring. In *International Conference on Ubiquitous Computing*, pages 56–72. Springer, 2005.

[Bat07] Jorge Batista. A drowsiness and point of attention monitoring system for driver vigilance. In *2007 IEEE Intelligent Transportation Systems Conference*, pages 702–708. IEEE, 2007.

[Bea05] Mary Beagon. *The Elder Pliny on the Human Animal: Natural History Book 7*. OUP Oxford, 2005.

[BHK+05] LI Bouwman, GJ Hiddink, MA Koelen, MJJAA Korthals, P Van't Veer, and C Van Woerkum. Personalized nutrition communication through ict application: how to overcome the gap between potential effectiveness and reality. *European journal of clinical nutrition*, 59(1):S108–S116, 2005.

[BMJZOV17] Colin Bellinger, Mohomed Shazan Mohomed Jabbar, Osmar Zaïane, and Alvaro Osornio-Vargas. A systematic review of data mining and machine learning for air pollution epidemiology. *BMC public health*, 17:1–19, 2017.

[BMLG16] Diego Zamora Blandón, John Edison Muñoz, David Sebastian Lopez, and Oscar Henao Gallo. Influence of a bci neurofeedback videogame

in children with adhd. quantifying the brain activity through an eeg signal processing dedicated toolbox. In *2016 IEEE 11th Colombian Computing Conference (CCC)*, pages 1–8. IEEE, 2016.

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BRJ17] Vieira Bruno, Silva Resende, and Cui Juan. A survey on automated food monitoring and dietary management systems. *Journal of health & medical informatics*, 8(3), 2017.

[CBCF15] Felicia Cordeiro, Elizabeth Bales, Erin Cherry, and James Fogarty. Rethinking the mobile food journal: Exploring opportunities for lightweight photo-based capture. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3207–3216, 2015.

[CCCL08] Pei-Yu Peggy Chi, Jen-Hao Chen, Hao-Hua Chu, and Jin-Ling Lo. Enabling calorie-aware cooking in a smart kitchen. In *International conference on persuasive technology*, pages 116–127. Springer, 2008.

[CET+15] Felicia Cordeiro, Daniel A Epstein, Edison Thomaz, Elizabeth Bales, Arvind K Jagannathan, Gregory D Abowd, and James Fogarty. Barriers and negative nudges: Exploring challenges in food journaling. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, pages 1159–1162, 2015.

[cho] Choregraphe. `http://doc.aldebaran.com/1-14/software/choregraphe/choregraphe_overview.html`. Accessed: 14-12-2022.

[CL14] Xue-Wen Chen and Xiaotong Lin. Big data deep learning: challenges and perspectives. *IEEE access*, 2:514–525, 2014.

[CMK18] Jacky Casas, Elena Mugellini, and Omar Abou Khaled. Food diary coaching chatbot. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 1676–1680, 2018.

[COC] Coco dataset. `http://cocodataset.org`. Accessed: 14-12-2022.

[col] Colaboratory. https://colab.research.google.com. Accessed: 14-12-2022.

[CRP] Convention on the rights of persons with disabilities. https://www.un.org/development/desa/disabilities/convention-on-the-rights-of-persons-with-disabilities.html. Accessed: 14-12-2022.

[CST+00] Nello Cristianini, John Shawe-Taylor, et al. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[CT65] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

[CTN18] Daniel Canedo, Alina Trifan, and António JR Neves. Monitoring students' attention in a classroom through computer vision. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 371–378. Springer, 2018.

[CWL+16] Shuiyuan Cheng, Gang Wang, Jianlei Lang, Wei Wen, Xiaoqi Wang, and Sen Yao. Characterization of volatile organic compounds from different cooking emissions. *Atmospheric Environment*, 145:299–307, 2016.

[CYH+18] Min Chen, Jun Yang, Long Hu, M Shamim Hossain, and Ghulam Muhammad. Urban healthcare big data system based on crowdsourced and cloud-based air quality indicators. *IEEE Communications Magazine*, 56(11):14–20, 2018.

[DAFRLG18] Roberto Díaz-Amador, Carlos A Ferrer-Riesgo, and Juan V Lorenzo-Ginori. Using image processing techniques and hd-emg for upper limb prosthesis gesture recognition. In *Iberoamerican Congress on Pattern Recognition*, pages 913–921. Springer, 2018.

[DHACN15] Kristen N DiFilippo, Wen-Hao Huang, Juan E Andrade, and Karen M Chapman-Novakofski. The use of mobile apps to im-

prove nutrition outcomes: a systematic literature review. *Journal of telemedicine and telecare*, 21(5):243–253, 2015.

[DKAW+21] Andac Demir, Toshiaki Koike-Akino, Ye Wang, Masaki Haruna, and Deniz Erdogmus. Eeg-gnn: Graph neural networks for classification of electroencephalogram (eeg) signals. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 1061–1067. IEEE, 2021.

[DN21] Yash Doshi and Divyanshi Nath. Designing a drone controller using electromyography signals. In *2021 International Conference on Communication information and Computing Technology (ICCICT)*, pages 1–6. IEEE, 2021.

[DSvEZ06] Gea Drost, Dick F Stegeman, Baziel GM van Engelen, and Machiel J Zwarts. Clinical applications of high-density surface emg: a systematic review. *Journal of Electromyography and Kinesiology*, 16(6):586–602, 2006.

[EJBS17] Luis A Estrada Jiménez, Marco E Benalcázar, and Nelson Sotomayor. Gesture recognition and machine learning applied to sign language translation. In *VII Latin American Congress on Biomedical Engineering CLAIB 2016, Bucaramanga, Santander, Colombia, October 26th-28th, 2016*, pages 233–236. Springer, 2017.

[fDCC+94] Centers for Disease Control, Prevention (CDC, et al. Populations at risk from particulate air pollution–united states, 1992. *MMWR. Morbidity and mortality weekly report*, 43(16):290–293, 1994.

[FGWG05] Katherine M Flegal, Barry I Graubard, David F Williamson, and Mitchell H Gail. Excess deaths associated with underweight, overweight, and obesity. *Jama*, 293(15):1861–1867, 2005.

[FHW+20] Chaoming Fang, Bowei He, Yixuan Wang, Jin Cao, and Shuo Gao. Emg-centered multisensory based technologies for pattern recognition in rehabilitation: state of the art and challenges. *Biosensors*, 10(8):85, 2020.

[Foo20] David Foord. Changes in technologies and meanings of upper limb prosthetics: Part i-from ancient egypt to early modern europe. In *MEC20 Symposium*, 2020.

[Fri72] Lawrence W Friedmann. Amputations and prostheses in primitive cultures. *Bulletin of prosthetics research*, 10(17):105–138, 1972.

[Fur05] Lydia Furman. What is attention-deficit hyperactivity disorder (adhd)? *Journal of child neurology*, 20(12):994–1002, 2005.

[gre] Epa sources greenhouse gas emissions. https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions. Accessed: 14-12-2022.

[GSS+19] Fatemeh Noushin Golabchi, Stefano Sapienza, Giacomo Severini, Phil Reaston, Frank Tomecek, Danilo Demarchi, MaryRose Reaston, and Paolo Bonato. Assessing aberrant muscle activity patterns via the analysis of surface emg data collected during a functional evaluation. *BMC musculoskeletal disorders*, 20(1):1–15, 2019.

[Gui41] Joy P Guilford. The phi coefficient and chi square as indices of item validity. *Psychometrika*, 6(1):11–19, 1941.

[HALI20] Hussein F Hassan, Sadiq J Abou-Loukh, and Ibraheem Kasim Ibraheem. Teleoperated robotic arm movement using electromyography signal with wearable myo armband. *Journal of King Saud University-Engineering Sciences*, 32(6):378–387, 2020.

[HFH+09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[HGS+08] Jack F Hollis, Christina M Gullion, Victor J Stevens, Phillip J Brantley, Lawrence J Appel, Jamy D Ard, Catherine M Champagne, Arlene Dalcin, Thomas P Erlinger, Kristine Funk, et al. Weight loss during the intensive intervention phase of the weight-loss maintenance trial. *American journal of preventive medicine*, 35(2):118–126, 2008.

[HGS16] Graham F Healy, Cathal Gurrin, and Alan F Smeaton. Informed perspectives on human annotation using neural signals. In *International Conference on Multimedia Modeling*, pages 315–327. Springer, 2016.

[HHP87] Richard W Homan, John Herman, and Phillip Purdy. Cerebral location of international 10–20 system electrode placement. *Electroencephalography and clinical neurophysiology*, 66(4):376–382, 1987.

[HYL17] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[HZG+09] Brahim Hamadicharef, Haihong Zhang, Cuntai Guan, Chuanchu Wang, Kok Soon Phua, Keng Peng Tee, and Kai Keng Ang. Learning eeg-based spectral-spatial patterns for attention level measurement. In *2009 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1465–1468. IEEE, 2009.

[HZLK08] He Huang, Ping Zhou, Guanglin Li, and Todd A Kuiken. An analysis of emg electrode configuration for targeted muscle reinnervation based neural machine interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(1):37–45, 2008.

[ICF] International classification of functioning, disability and health. https://www.who.int/standards/classifications/international-classification-of-functioning-disability-and-health. Accessed: 14-12-2022.

[IK20] Lina Elsherif Ismail and Waldemar Karwowski. Applications of eeg indices for the quantification of human cognitive performance: A systematic review and bibliometric analysis. *Plos one*, 15(12):e0242857, 2020.

[IRT20] Ditsuhi Iskandaryan, Francisco Ramos, and Sergio Trilles. Air quality prediction in smart cities using machine learning technologies based on sensor data: a review. *Applied Sciences*, 10(7):2401, 2020.

[JBR+17] Utkarshani Jaimini, Tanvi Banerjee, William Romine, Krishnaprasad Thirunarayan, Amit Sheth, and Maninder Kalra. Investigation of an indoor air quality sensor for asthma management in children. *IEEE sensors letters*, 1(2):1–4, 2017.

[JJ20] Dong-Hwa Jeong and Jaeseung Jeong. In-ear eeg based attention state classification using echo state network. *Brain sciences*, 10(6):321, 2020.

[JR19] Hanadi Abbas Jaber and Mofeed Turky Rashid. Hd-semg gestures recognition by svm classifier for controlling prosthesis. *Iraqi Journal of Computers, Communications, Control and System Engineering (IJCCCE)*, 19(1):10–19, 2019.

[JSZ+22] Yujian Jiang, Lin Song, Junming Zhang, Yang Song, and Ming Yan. Multi-category gesture recognition modeling based on semg and imu signals. *Sensors*, 22(15):5855, 2022.

[JYBMM20] Andrés Jaramillo-Yánez, Marco E Benalcázar, and Elisa Mena-Maldonado. Real-time hand gesture recognition using surface electromyography and machine learning: a systematic literature review. *Sensors*, 20(9):2467, 2020.

[ker] Keras. `https://keras.io/`. Accessed: 14-12-2022.

[KKAJN18] Rami N Khushaba, Agamemnon Krasoulis, Adel Al-Jumaily, and Kianoush Nazarpour. Spatio-temporal inertial measurements feature extraction improves hand movement pattern recognition without electromyography. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2108–2111. IEEE, 2018.

[KN21] Rami N Khushaba and Kianoush Nazarpour. Decoding hd-emg signals for myoelectric control-how small can the analysis window size be? *IEEE Robotics and Automation Letters*, 6(4):8569–8574, 2021.

[LAE+16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single

shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[LBC+18] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

[LCC13] Ning-Han Liu, Cheng-Yu Chiang, and Hsuan-Chin Chu. Recognizing the degree of human attention using eeg signals from mobile sensors. *sensors*, 13(8):10273–10286, 2013.

[LGSMV14] Miguel Angel Lopez-Gordo, Daniel Sanchez-Morillo, and F Pelayo Valle. Dry eeg electrodes. *Sensors*, 14(7):12847–12870, 2014.

[LLZL18] Kai Lukoff, Taoxi Li, Yuan Zhuang, and Brian Y Lim. Tablechat: mobile food journaling to facilitate family support for healthy eating. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):1–28, 2018.

[LQQ+18] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018.

[LSK10] Guanglin Li, Aimee E Schultz, and Todd A Kuiken. Quantifying pattern recognition—based myoelectric control of multifunctional transradial prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):185–192, 2010.

[LZX+13] Songpo Li, Jiucai Zhang, Linting Xue, Fernando J Kim, and Xiaoli Zhang. Attention-aware robotic laparoscope for human-robot cooperative surgery. In *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 792–797. IEEE, 2013.

[MA19] Prathusha Maduri and Hossein Akhondi. Upper limb amputation. 2019.

[Mar71] Jean W Marr. Individual dietary surveys: purposes and methods. *World review of nutrition and dietetics*, 13:105–164, 1971.

[MCFdR19] Fabiana SV Machado, Wagner D Casagrande, Anselmo Frizera, and Flavia EM da Rocha. Development of serious games for neurorehabilitation of children with attention-deficit/hyperactivity disorder through neurofeedback. In *2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, pages 91–97. IEEE, 2019.

[MCR10] Silvestro Micera, Jacopo Carpaneto, and Stanisa Raspopovic. Control of hand prostheses using peripheral information. *IEEE reviews in biomedical engineering*, 3:48–68, 2010.

[MDCL+17] Radhika Menon, Gaetano Di Caterina, Heba Lakany, Lykourgos Petropoulakis, Bernard A Conway, and John J Soraghan. Study on interaction between temporal and spatial information in classification of emg signals for myoelectric prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1832–1842, 2017.

[MFE+16] Alessandra Moschetti, Laura Fiorini, Dario Esposito, Paolo Dario, and Filippo Cavallo. Recognition of daily gestures with wearable inertial rings and bracelets. *Sensors*, 16(8):1341, 2016.

[MHH+02] Jennifer Mankoff, Gary Hsieh, Ho Chak Hung, Sharon Lee, and Elizabeth Nitao. Using low-cost sensing to support nutritional awareness. In *International conference on ubiquitous computing*, pages 371–378. Springer, 2002.

[min] Mind monitor. `https://mind-monitor.com`. Accessed: 14-12-2022.

[MM20] Silvia Maria Massa and Marco Manolo Manca. Toward a brain-controlled prosthetic arm through advanced machine learning methods (short paper). In *SmartPhil@ IUI*, pages 50–58, 2020.

[MMK06] Lena Mamykina, Elizabeth D Mynatt, and David R Kaufman. Investigating health management practices of individuals with diabetes. In

Proceedings of the SIGCHI conference on Human Factors in computing systems, pages 927–936, 2006.

[mor] More itertools. https://more-itertools.readthedocs.io/en/latest/. Accessed: 14-12-2022.

[MOS+21] Nebojša Malešević, Alexander Olsson, Paulina Sager, Elin Andersson, Christian Cipriani, Marco Controzzi, Anders Björkman, and Christian Antfolk. A database of high-density surface electromyogram signals comprising 65 isometric hand gestures. *Scientific Data*, 8(1):1–10, 2021.

[MRBL11] Marcus R Munafò, Nicole Roberts, Linda Bauld, and Ute Leonards. Plain packaging increases visual attention to health warnings on cigarette packs in non-smokers and weekly smokers but not daily smokers. *Addiction*, 106(8):1505–1510, 2011.

[mus] Muse sensors. https://choosemuse.com/. Accessed: 12-10-2022.

[nua] Nuance. https://www.nuance.com. Accessed: 14-12-2022.

[Nus81] Henri J Nussbaumer. The fast fourier transform. In *Fast Fourier Transform and Convolution Algorithms*, pages 80–111. Springer, 1981.

[O+08] World Health Organization et al. *The global burden of disease: 2004 update*. World Health Organization, 2008.

[O+11] World Health Organization et al. World report on disability 2011: World health organization, 2011.

[O+16] World Health Organization et al. Ambient air pollution: A global assessment of exposure and burden of disease. 2016.

[O+19] World Health Organization et al. Attention deficit hyperactivity disorder (adhd). Technical report, World Health Organization. Regional Office for the Eastern Mediterranean, 2019.

[oESA22] United Nations Department of Economic and Population Division Social Affairs. World population prospects 2022: Summary of results. un desa/pop/2022/tr/no. 3. 2022.

[OLKT17] Yasuyuki Ochi, Tassaneewan Laksanasopin, Boonserm Kaewkamnerdpong, and Kejkaew Thanasuan. Neurofeedback game for attention training in adults. In *2017 10th Biomedical Engineering International Conference (BMEiCON)*, pages 1–5. IEEE, 2017.

[ONSJ18] Hyungik Oh, Jonathan Nguyen, Soundarya Soundararajan, and Ramesh Jain. Multimodal food journaling. In *Proceedings of the 3rd International Workshop on Multimedia for Personal Health and Health Care*, pages 39–47, 2018.

[Ott21] Andreas Otte. Artifacts: Gottfried "götz" von berlichingen—the "iron hand" of the renaissance. *Clinical Orthopaedics and Related Research®*, 479(1):210–211, 2021.

[PCJLGS+20] Inmaculada Penuelas-Calvo, Lin Ke Jiang-Lin, Braulio Girela-Serrano, David Delgado-Gomez, Rocio Navarro-Jimenez, Enrique Baca-Garcia, and Alejandro Porras-Segovia. Video games for the assessment and treatment of attention-deficit/hyperactivity disorder: a systematic review. *European child & adolescent psychiatry*, pages 1–16, 2020.

[PCP+21] Nikolaos Peladarinos, Vasileios Cheimaras, Dimitrios Piromalis, Konstantinos G Arvanitis, Panagiotis Papageorgas, Nikolaos Monios, Ioannis Dogas, Milos Stojmenovic, and Georgios Tsaramirsis. Early warning systems for covid-19 infections based on low-cost indoor air-quality sensors and lpwans. *Sensors*, 21(18):6183, 2021.

[PS15] Pramod Kumar Pisharady and Martin Saerbeck. Recent methods and databases in vision-based hand gesture recognition: A review. *Computer Vision and Image Understanding*, 141:152–165, 2015.

[PSB+19] Nawadita Parajuli, Neethu Sreenivasan, Paolo Bifulco, Mario Cesarelli, Sergio Savino, Vincenzo Niola, Daniele Esposito, Tara J Hamilton, Ganesh R Naik, Upul Gunawardana, et al. Real-time

emg based pattern recognition control for hand prostheses: a review on existing methods, challenges and future implementation. *Sensors*, 19(20):4596, 2019.

[PSRJ17] Viral Parekh, Ramanathan Subramanian, Dipanjan Roy, and CV Jawahar. An eeg-based image annotation system. In *National Conference on Computer Vision, Pattern Recognition, Image Processing, and Graphics*, pages 303–313. Springer, 2017.

[Put05] V Putti. Historical prostheses. *Journal of Hand Surgery*, 30(3):310–325, 2005.

[PyG] Pytorch geometric. `https://pytorch-geometric.readthedocs.io/en/latest/index.html`. Accessed: 14-12-2022.

[QT09] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE conference on computer vision and pattern recognition*, pages 413–420. IEEE, 2009.

[RSNB94] MJPF Ritt, PR Stuart, L Naggar, and RD Beckenbaugh. The early history of arthroplasty of the wrist from amputation to total wrist implant. *The Journal of Hand Surgery: British & European Volume*, 19(6):778–782, 1994.

[RTMH18] Francisco Ramos, Sergio Trilles, Andrés Muñoz, and Joaquín Huerta. Promoting pollution-free routes in smart cities using air quality sensor networks. *Sensors*, 18(8):2507, 2018.

[RVCM20] Susanna Rampichini, Taian Martins Vieira, Paolo Castiglioni, and Giampiero Merati. Complexity analysis of surface electromyography for assessing the myoelectric manifestation of muscle fatigue: A review. *Entropy*, 22(5):529, 2020.

[RWJ+21] Jacob Rantas, David Wang, Will Jarrard, James Sterchi, Alan Wang, Mahsa Pahlavikhah Varnosfaderani, and Arsalan Heydarian. A user interface informing medical staff on continuous indoor environmental quality to support patient care and airborne disease mitigation.

In *2021 Systems and Information Engineering Design Symposium (SIEDS)*, pages 1–6. IEEE, 2021.

[sci] Scikit-learn. https://scikit-learn.org/stable/. Accessed: 14-12-2022.

[SE11] Erik Scheme and Kevin Englehart. Electromyogram pattern recognition for control of powered upper-limb prostheses: state of the art and challenges for clinical use. *Journal of Rehabilitation Research & Development*, 48(6), 2011.

[She64] E David Sherman. A russian bioelectric-controlled prosthesis: Report of a research team from the rehabilitation institute of montreal. *Canadian Medical Association Journal*, 91(24):1268, 1964.

[SHLK10] Lauren H Smith, Levi J Hargrove, Blair A Lock, and Todd A Kuiken. Determining the optimal window length for pattern recognition-based myoelectric control: balancing the competing effects of classification error and controller delay. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(2):186–192, 2010.

[SHP+18] Agnes Sturma, Laura A Hruby, Cosima Prahm, Johannes A Mayer, and Oskar C Aszmann. Rehabilitation of upper extremity nerve injuries using surface emg biofeedback: protocols for clinical application. *Frontiers in neuroscience*, 12:906, 2018.

[SIA+15] Sean Semple, Azmina Engku Ibrahim, Andrew Apsley, Markus Steiner, and Stephen Turner. Using a new, low-cost air quality sensor to quantify second-hand smoke (shs) levels in homes. *Tobacco control*, 24(2):153–158, 2015.

[SLT+18] Wan-Ting Shi, Zong-Jhe Lyu, Shih-Tsang Tang, Tsorng-Lin Chia, and Chia-Yen Yang. A bionic hand controlled by hand gesture recognition based on surface emg signals: A preliminary study. *Biocybernetics and Biomedical Engineering*, 38(1):126–135, 2018.

[SNF14] Antonietta Stango, Francesco Negro, and Dario Farina. Spatial correlation of high density emg signals provides features robust to

electrode number and shift in pattern recognition for myocontrol. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(2):189–198, 2014.

[SRR20] Raza Abdulla Saeed, Diego Reforgiato Recupero, and Paolo Remagnino. A boundary node method for path planning of mobile robots. *Robotics and autonomous systems*, 123:103320, 2020.

[SS15] Celal Savur and Ferat Sahin. Real-time american sign language recognition system using surface emg signal. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 497–502. IEEE, 2015.

[SS16] Celal Savur and Ferat Sahin. American sign language recognition system by using surface emg signal. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 002872–002877. IEEE, 2016.

[SSM+18] Sougata Sen, Vigneshwaran Subbaraju, Archan Misra, Rajesh Balan, and Youngki Lee. Annapurna: building a real-world smartwatch-based automated food journal. In *2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*, pages 1–6. IEEE, 2018.

[Sun23] D. Sundararajan. *The Discrete Fourier Transform*, pages 125–160. Springer Nature Switzerland, Cham, 2023.

[T+02] Michal Teplan et al. Fundamentals of eeg measurement. *Measurement science review*, 2(2):1–11, 2002.

[TBC+20] Hatice Tankisi, David Burke, Liying Cui, Mamede de Carvalho, Satoshi Kuwabara, Sanjeev D Nandedkar, Seward Rutkove, Erik Stålberg, Michel JAM van Putten, and Anders Fuglsang-Frederiksen. Standards of instrumentation of emg. *Clinical neurophysiology*, 131(1):243–258, 2020.

[TBW+18] Emma L Tonkin, Alison Burrows, Przemysław R Woznowski, Pawel Laskowski, Kristina Y Yordanova, Niall Twomey, and Ian J Crad-

dock. Talk, text, tag? understanding self-annotation of smart home data from a user's perspective. *Sensors*, 18(7):2365, 2018.

[ten] Tensorflow object detection api. https://bit.ly/2lPqHJk. Accessed: 14-12-2022.

[uHo] Uhoo sensors. https://getuhoo.com/. Accessed: 14-12-2022.

[VHS10] Vivianne HM Visschers, Rebecca Hess, and Michael Siegrist. Health motivation and product design determine consumers' visual attention to nutrition information on food products. *Public health nutrition*, 13(7):1099–1106, 2010.

[wat] Ibm watson speech to text. https://www.ibm.com/cloud/watson-speech-to-text. Accessed: 14-12-2022.

[WG07] Mary H Wilde and Suzanne Garvin. A concept analysis of self-monitoring. *Journal of advanced nursing*, 57(3):339–350, 2007.

[Whe03] Kevin R Wheeler. Device control using gestures sensed from emg. In *Proceedings of the 2003 IEEE International Workshop on Soft Computing in Industrial Applications, 2003. SMCia/03.*, pages 21–26. IEEE, 2003.

[WHO] Who malnutrition. https://www.who.int/news-room/fact-sheets/detail/malnutrition. Accessed: 14-12-2022.

[WKS+14] R Williams, Vasu Kilaru, E Snyder, A Kaufman, T Dye, A Rutter, A Russell, and H Hafner. Air sensor guidebook. us environmental protection agency, washington, dc. Technical report, EPA/600/R-14/159 (NTIS PB2015-100610), 2014.

[WMJ+15] Pierre Wargnier, Adrien Malaisé, Julien Jacquemot, Samuel Benveniste, Pierre Jouvelot, Maribel Pino, and Anne-Sophie Rigaud. Towards attention monitoring of older adults with cognitive impairment during interaction with an embodied conversational agent. In *2015 3rd IEEE VR International Workshop on Virtual and Augmented Assistive Technology (VAAT)*, pages 23–28. IEEE, 2015.

[WST+22] Simon Wein, Alina Schüller, Ana Maria Tomé, Wilhelm M Malloni, Mark W Greenlee, and Elmar W Lang. Forecasting brain activity based on models of spatiotemporal brain dynamics: A comparison of graph neural network architectures. *Network Neuroscience*, 6(3):665–701, 2022.

[WYH+21] Zitong Wan, Rui Yang, Mengjie Huang, Nianyin Zeng, and Xiaohui Liu. A review on transfer learning in eeg signal analysis. *Neurocomputing*, 421:1–14, 2021.

[YLW+19] Kristina Yordanova, Stefan Lüdtke, Samuel Whitehouse, Frank Krüger, Adeline Paiement, Majid Mirmehdi, Ian Craddock, and Thomas Kirste. Analysing cooking behaviour in home settings: Towards health monitoring. *Sensors*, 19(3):646, 2019.

[YWP+17] Kristina Yordanova, Samuel Whitehouse, Adeline Paiement, Majid Mirmehdi, Thomas Kirste, and Ian Craddock. What's cooking and why? behaviour recognition during unscripted cooking tasks for health monitoring. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom Workshops)*, pages 18–21. IEEE, 2017.

[ZBW+10] Fengqing Zhu, Marc Bosch, Insoo Woo, SungYe Kim, Carol J Boushey, David S Ebert, and Edward J Delp. The use of mobile devices in aiding dietary assessment and evaluation. *IEEE journal of selected topics in signal processing*, 4(4):756–766, 2010.

[ZCH+20] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020.

[ZLL+21] Yuanyuan Zheng, Rongyang Li, Sha Li, Yudong Zhang, Shunkun Yang, and Huansheng Ning. A review on serious games for adhd. *arXiv preprint arXiv:2105.02970*, 2021.

[ZO14] Kevin J Zuo and Jaret L Olson. The evolution of functional hand replacement: From iron prostheses to hand transplantation. *Plastic Surgery*, 22(1):44–51, 2014.

[zor] Zora robotics. https://www.zorarobotics.be/robot-family. Accessed: 14-12-2022.

[ZS20] He Zhang and Ravi Srinivasan. A systematic review of air quality sensors, guidelines, and measurement studies for indoor air quality management. *Sustainability*, 12(21):9045, 2020.