FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

U. PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Cooperative Human-Machine Interaction in Industrial Environments

Liliana Patricia Saldanha Antão

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Gil Manuel Gonçalves

Co-supervisor: Rui Pinto

Co-supervisor: João Reis

July 24, 2017

Abstract

Despite the existence of various solutions in the industrial domain for cooperation between robots and humans, they tend to focus mainly on safety issues with very few advances in the adaptation of industrial equipment to the characteristics of the operator and his way of working. For several years, adaptation in a human-robot collaboration environment was single sided, as only the operator adapts his working operations facing the robot characteristics, which leads to high levels of stress and fatigue of the human operator. Nowadays, the paradigm is changing towards the adaptation of human operator to industrial equipment and vise versa. The adaptation of a robot to the human is achieved by enabling the machine to learn the physical and psychological characteristics of each operator, in order to create a working profile for each individual. Thus, the main objective is to analyze the relationship between human operators and robots in an industrial environment, and therefore explore human-machine collaboration by correlating sensorial data from all the entities involved in the process. With this in mind, by performing sensor fusion and data analysis representing actions and biometric signals from the human operator, industrial robots will be empowered of self-adaptation capabilities.

In this dissertation, an industrial collaborative environment is achieved using a Cyber-Physical Production System (CPPS). This CPPS consists in three main parts, namely sensing and actuating equipment, logical entities called Smart Components and a Cloud infrastructure. Sensing devices are based on biometric sensors - BITalino's ECG and EDA - and a vision system - Kinect - in order to monitor the human operator working profile. A robotic arm is used as actuating device. Each equipment is virtualized into an agent-based representation, based on the Smart Component concept, which communicate sensor data with a Cloud infrastructure responsible for data processing and decision making. Sensor data is analyzed in order to infer levels of stress and fatigue through a fuzzy logic system. Decision making is based on the MAPE-K architecture, enabling the robotic arm self-adaptation. Results from human subject tests are presented here to validate the proposed methodology, proving that the system can detect stress with an accuracy of 77,6% and fatigue with an accuracy of 70%, as well as detect the subject's position and movement with a true positive rate of 70,7%. Facing the movement and levels of stress and fatigue of the human operator, the robotic arm should be able to change autonomously it's task execution, namely speed of its movement and the correct operation according to the habits of the operator.

Acknowledgements

First of all I would like to thank Professor Gil Manuel Gonçalves, my supervisor, for the guidance and follow-up of the work done, for all the suggestions and corrections, and for giving me the basis for this work to be possible.

The biggest thanks to Rui Pinto and João Reis, for being the co-supervisors every student hopes for and even more, for guiding me and helping me in every occasion with the greatest patience in the world and doing all that with a good sense of humor. Without you, this dissertation would not be half of what it is.

A big thank you to Catarina Simões, Rita Rodrigues, Marta Costa, Ricardo Silva, João Sá and Héber Sobreira for always trying to make me still have a social life. To Eva Pereira and Marco Rodrigues, a huge thanks for putting up with me every time I freaked out (which was almost every day), for giving me advice in every call and every coffee. Without you all, I would have possibly gone crazy.

To my boyfriend, for understanding all the nights spent in front of the laptop writing, for dealing with my bad mood and tiredness, for calming me when I started to freak out with all the work, and still always having some bad joke to try make me laugh, a super saiyan thank you. Without you I would probably be stuck in a endless bad mood, thank you so much for everything, you are the best in the whole world.

My final and greatest thanks of all to my parents and little bro for all the support through all of this years, you made unthinkable efforts to make this journey possible, remanding me every step of the way that I had all the capabilities to be whatever I wanted to be. You are the best parents anyone could have, and although I'm very proud to become an engineer nothing makes me more proud than being your daughter. And you little João, sorry for all the times I kicked you out of my room while I was writing, now that it's over I will return to make fun of you instead of throwing you out, love you little bro.

Liliana Antão

“I have not failed
I’ve just found 10 000 ways that doesn’t work”

Thomas Edison

Contents

List of Figures

List of Tables

Abbreviations

ACC	Accelerometer
ANS	Autonomous Nervous System
API	Application Programming Interface
AUROC	Area Under the Receiver Operating Characteristic curve
BN	Bayesian Network
BVP	Blood Volume Pulse
CC	Cyber Component
CHMI	Cooperative Human-Machine Interaction
CMfg	Cloud Manufacturing
CPPS	Cyber-Physical Production System
CPS	Cyber-Physical System
ECG	Electrocardiogram
EDA	Electrodermal Activity
EEG	Electroencephalogram
EMG	Electromyography
EOG	Electrooculography
FIR	Finite Impulse Response
GSR	Galvanic Skin Response
HC	Human Component
HMC	Human-Machine Collaboration
HMI	Human-Machine Interaction
HMM	Hidden Markov Model
HR	Heart-Rate
HRI	Human-Robot Interaction
HRV	Heart-Rate Variability
IBI	Inter-Beat Interval
IIoT	Industrial Internet of Things
IoT	Internet of Things
kNN	kth Nearest Neighbors
KPI	Key Performance Indicator
LUX	Light sensor
LED	Light Emitting Diode
MAPE	Monitor-Analyze-Plan-Execute
MAPE-K	Monitor-Analyze-Plan-Execute-Knowledge
MCU	Micro-Controller Unit
NN	Neural Networks
PD	Pupil Diameter
PLC	Programmable Logic Controller

PNS	Parasympathetic Nervous System
PPG	Photoplethysmograph
PRV	Pulse Rte Variability
PSD	Power Spectral Density
ROC	Receiver Operating Characteristic
SCL	Skin Conductance Level
SCR	Skin Conductance Response
SDK	Software Development Kit
SNR	Signal-to-Noise Ratio
SNS	Sympathetic Nervous System
SOA	Service Oriented Architecture
ST	Skin Temperature
SVM	Support Vector Machine
ToF	Time-Of-Flight
VR	Virtual Reality
WIP	Work In Progress

Chapter 1

Introduction

1.1 Context

Nowadays, most of the manufacturing processes are characterized by the collaboration between machines and human operators, where the machines are responsible for executing the heavier and/or repetitive operations, while the human operators are responsible for handling shop-floor equipment and supervising processes for high-level decision making. In a typical collaborative environment, both human operators and machines execute operations in shared tasks. This collaboration makes the production process faster and more efficient while maintaining the quality of the end product within standards.

Usually, this close collaboration between both parts demands an adaptation from the human operator, mainly because the operator needs to adjust his actions accordingly to the machine's running level. For instance, if in an assembly line exists one machine with a high level of strain, it will become the *bottleneck* of the line. This machine will require a doubled effort from the operator to compensate its throughput regarding the others and avoid the accumulation of *Work In Progress (WIP)*. However, the human operator can be viewed as one resource in the production line that might experience outworn the same way as a machine does. The difference is that the human operator outworn occurs in the form of stress and/or fatigue. With this in mind, the collaboration between human operator and machine should be bilateral, as the machine should also adapt to the human operator, be aware of the operator's state and be prepared to compensate in case of human flaw.

Currently, the research work in the area of Human-Machine Collaboration (HMC) is mainly focused on matters of operators' safety. Novel approaches focus on stopping or movement reversion methods for the industrial robot if in a dangerous situation, equipping the robot with several internal and external sensors. Faced with this situation, there is the need to develop collaborative methodologies, providing emphasis on the adaptation of the industrial equipment towards the human operators, on which the robots have to be aware of the physical and psychological characteristics of each operator, creating a work profile, that will vary between the different workers.

1.2 Motivation

Although some solutions regarding collaboration between human operators and industrial equipment already exist, few advances have been made in the matter of equipment's adaptation to the operators' characteristics and way of acting in the production process. Therefore, the major motivation of the research work developed in this dissertation is to contribute with a novel approach in the area of HMC. The present approach focus in the adaptation of the machine facing the working profile of the human operator, such as speed of his task executions or the order preference of operations during the task execution. This adaptation is achieved through machine learning techniques, by understanding the operator's working habits. The machine learning process is based on sensor fusion and data analysis of representative information regarding biometric signals and movement from the operator. The learning process results in the equipment capability to infer the operator's stress and fatigue levels. Based on this, the equipment will be capable of adapting itself, not only making up failures that would harm the manufacturing process but also try to improve the operator's working conditions.

1.3 Objectives

This dissertation has as main objective the analysis of the relationship between human operators and industrial equipment when performing a task in a collaborative environment. This collaborative environment will be achieved by modeling the problem and building a Cyber-Physical Production System (CPPS). The test case scenario consists of an operator equipped with several biometric sensors, a robotic arm with external sensors and a Cloud infrastructure to analyze and process data.

With this analysis, it is intended to explore collaborative processes between man and machine, analyzing and correlating relevant sensory data from the operator and from the robotic arm. The collected sensory data allows, not only to understand habits and ways of executing the operator's tasks by creating a work profile but also to infer his stress and fatigue levels over time. The stress and fatigue information will be used to introduce self-adaptation abilities into the robotic arm. This will, not only allow to minimize human flaws impact in the overall process but also to improve the worker's conditions during manufacturing processes, without affecting product's quality and process efficiency.

1.4 Dissertation Structure

This document is divided into six main chapters. Chapter 2 presents the problem of Cooperative Human-Machine Interaction (CHMI) in industrial environments, addressing the current HMI reality. Moreover, it defines the paradigm of CPPS and self-adaptive systems, which are the key enabling for new industrial businesses opportunities, such as CHMI. Chapter 3 refers to the literature regarding CHMI, focusing on the human operator monitoring strategies, methods, devices, and metrics used for monitoring human operator's performance, as well as systems that facilitate this monitoring. Chapter 4 describes the proposed solution for the CHMI problem, detailing the solution implementation and development. Chapter 5 validates the proposed solution and explains the tests conducted to evaluate its performance, revealing and discussing its results. Finally, Chapter 6 discusses and identifies conclusions and possible future work regarding the developed work.

Chapter 2

Cooperative Human-Machine Interaction

In this chapter, the CHMI problem will be defined in the reality of today's industrial environments, by presenting the Human Machine Interaction (HMI) paradigm and its three main variations - safety, human-machine interfaces and Human-Machine Collaboration (HMC). Also, the importance of Cyber-Physical Systems (CPS) in the manufacturing context is presented, analyzing its concepts and architectures, and introducing the particularities of the application of CPS in production systems - CPPS. Moreover, and since CPS are by nature, self-adaptive systems, the well known MAPE-K reference model for self-adaptation is also explained. Finally, the scope of the proposed solution is presented.

2.1 Human-Machine Interaction

The current industrial production paradigm is concentrated with large scale manufacturing industries that have high levels of automation. These automatic processes for collaborative assembly are composed of human and non-human elements, where most of the machines, as robotic manipulators, are used in industrial environments on account of their precision, repeatability, and durability. They are normally applied in repetitive assembly tasks that can be efficiently performed by machines, where the production system rarely changes, only being necessary to control the position of the robot. But with the arising of new manufacturing realities, such as multi human – multi robot stations, the urgency for HMI solutions in the industrial environment has emerged, obliging to a smooth integration of humans and robots in the same production line [8].

This integration should not lead to completely automated systems, where the machines perform most of the tasks, nor should become fully-manual, where the human takes the lead role in task execution. Instead, this integration should be characterized by a cooperative interaction between the two parts, seeing the machine as the co-worker to the operator and complementing each other in processes that require several tasks. Ultimately, by joining the machine's resilience with the human's overall knowledge, will result in the improvement of task execution efficiency.

Until the present days in industrial environments, there has been little advances in the relation between the shop-floor operator and the machines executing the manufacturing process. Therefore, a step forward to a more advanced interaction between machine and operator should be taken, towards a more adaptive and rich symbiosis. In order to achieve this kind of interaction, industrial robots should be able to interact safely, intuitively and appropriately with operators, other robots, and other systems within a production environment, responding to changes in that environment and including the ability to apply auto-configuration strategies.

When trying to achieve this human-machine interplay, a better interaction can be obtained by combining some aspects, such as safety parameters, human-machine interfaces and human-machine collaboration. This combination leads to an increase of the flexibility of industrial robots, enabling the cooperation with humans (including physical cooperation), and providing much faster and much more intuitive automation systems [8].

2.1.1 Safety

In a collaborative environment, the interaction between application robots and humans is mandatory, working in direct contact in a predefined space. So, as one would expect, the main requirement for these interactions is to be safe. So, most of the efforts in recent years are done to assure human safety. Until recently, the most common approach to guarantee the operator's safety, when it comes to HMI, is to isolate the machine (an industrial robot, for instance), by using physical barriers like fences. The industrial robot is required to stop completely in case of violation of the robot's workspace by a human worker. This can cause a great reduction in the productivity as it leads to interruptions and rebooting of processes, and it also prevents true cooperation between the two parts, restricting their potential and limiting the possible advantages that come from a complete HMI.

With the evolution of industrial processes and its collaborative necessities, new standards appeared that allow human and machine coexistence in the same space if the robot respected velocity limits (up to 250mm/s) and static force limits (150N) that prevented injuries in the operator [9], or allowing the robots movement only when controlled by a human. Another solutions encompassed human monitoring, using external sensors, to allow the robot to perceive when its co-worker is close, calculating his distance, and slowing down, altering paths, or even coming to a full stop if necessary. So to implement HMI in a safe way, pre-collision and/or post collision planning is mandatory. The most common solution for post collision is to reduce the contact force when the collision with the human is detected through torque sensors, while in pre-collision the robot's path is altered according to the distance to the human.

For the future of industrial processes, where a robot is able to safely collaborate with human operators, these safety mechanisms need to be adaptable, possibly configured by the operators with the use of interfaces instead of programmers, allowing a more specific solution for each situation and given context. With these new methods to ensure human safety without physically limiting the operator, CHMI can be achieved. This allows to freely use robots in industrial environments with a high density of human operators. Consequently, the manufacturing process's productivity increases by allowing to bring together the resilience of the human operator and the accuracy of the robot.

2.1.2 Human-Machine Interfaces

From an industrial perspective, human-machine interfaces have the advantage to allow the human operator to closely monitor production processes and respond to changing demands, providing an intuitive way to control or maintain a machine, ultimately improving efficiency and productivity. The traditional interface solutions involve devices as simple as computers with keypads, a mobile device or even the push of a button, that allow the shop-floor worker to interact with the machines in a very primitive way. These solutions were stand-alone, isolated terminals and only distributed by the original equipment manufacturer.

With the new demands of manufacturing processes, novel approaches were proposed, which despite their complexity, are still intuitive interfaces. For instance, Augmented Reality (AR) and Virtual Reality (VR) approaches can recreate the manufacturing process, serving as an interface between the user and the system, allowing to examine the production system [10]. With AR, using smartphones, tablets, and smartglasses, pertinent information from the process can be inserted in the worker's field of vision. This enhances, e.g., the process's maintenance, by supplying virtual instructions and monitoring the manufacturing operation through the system's status.

Other forms of HMI consist on speech and gesture recognition. Speech recognition brings advantages in cases where the worker's hands are occupied, allowing the use of voice commands to control the process. In hands-free or even eyes-free environments, the human operator can command through speech, supporting much greater quantities of information than other interfaces [11]. Gesture recognition solutions allow the user to assign an action to a machine/robot through gestures, which are captured by sensor networks or cameras. There are several solutions when it comes to cameras for gesture and action recognition: from rigorous and expensive systems like Optitrack, which requires the use of reflective markers for high precision track of movements, to low-cost solutions, such as the Kinect sensor, that rely on the computation of skeleton models.

As it can be concluded, interfaces can improve overall HMI in industrial environments. Besides allowing the user to see the system's overall information in an intuitive way, it also lets him send commands to the machines when a certain task is required to be executed. These new interfaces provide the user with an intuitive way to teach a robot's behavior or lead it to perform the desired task in assembly sequences, without the need of specialized programmers. It is fair to say that the existing solutions for industrial human-machine interfaces are evolving rapidly, being adjusted to the needs of the environment. But with the emerging of CPS in the industrial reality, these interfaces should also push and pull data to/from one or more Cloud infrastructure, enabling the integration of devices as smartphones or tablets, as well as being easily networked with other components on the plant floor.

2.1.3 Human-Machine Collaboration

Given today's reality in industrial processes described in Section 2.1, it is mandatory that the technologies and interaction aspects used in manufacturing widen its bases, in order to achieve the new emerging goals for production systems: increase flexibility of the industrial robots, faster and more intuitive configurations, and as usual, a more productive and efficient process. The use of industrial robots that can react to environmental changes can greatly improve the productivity of the manufacturing, creating a human machine symbioses, i.e, an HMC with the goal of compensating dynamic changes in the process where the collaboration exists.

In order to develop this sensibility, these robots usually need to take into consideration additional sensory data derived from monitoring changes in the production system. Sensor information enables them to adapt their behavior according to those changes, which can occur in the surrounding environment or with the worker in the collaboration environment. Availability in real-time regarding all relevant information of the environment and the operator is essential, as it is the basis for real-time adaptive motion planning, prediction, and control. This is especially beneficial in CPPS, where the sensory data from all the physical devices in the system could enable a fusion of all features, and consequently allow the machine to adapt its motion, compensating potential human flaws or adapt its actions, in order to motivate the worker to increase productivity [12].

The research work regarding HMC in an industrial environment it is in its infancy, lacking existent solutions that emphasize the industrial machine flexibility to the surrounding context. Moreover, very few approaches focus in the machine self-adaptation facing the human co-worker, where robots have knowledge of the operator's state and the surrounding environment, being prepared to compensate in case of decreased human efficiency or perform to improve the worker's conditions.

In general, most of the tasks in collaborative processes focus on holding an object for the operator and retrieving it on demand [13]. However Tsarouchi *et al.* [14] proposed a decision-making method that permits human-robot task allocation for the execution of sequential tasks, where body gestures are the means of HMI for commanding and guiding. Wang *et al.* [15] proposed an HMC system as part of the assembly process of an automotive engine, where some steps were active HMC actions, such as place and hold parts and tighten screws. The system also includes access to smartphones and tablets, while the operator assistance is provided by a private Cloud server. Moreover, within the European R&D project LOCOBOT, a toolkit for low-cost robots was developed from a set of plug-and-produce kinematic modules with intelligent sensing and actuating structures. The robot is equipped with a stereo camera system and audio components to obtain and process audio visual information so that it can learn to cooperate with human workers [8].

2.2 Smart Manufacturing

Nowadays, the demands of the global markets for rapid product development, characterized by high levels of customization (known as production oriented for mass customization) impose hard requirements to manufacturing companies, such as resilience, robustness and reconfigurable manufacturing processes. One of the proposed approaches to tackle these requirements consists in the development of agent-based systems, which allow efficient, flexible, reconfigurable and resilient overall plant control [16]. Agent-based systems enable the dynamic reconfiguration of the manufacturing system, in order to respond to changes in market demand for production [17]. This new desired characteristics of manufacturing control systems require that they operate decentralized of rigid control structures. Reconfigurability is one of the most important attributes, as it enables the system to respond quickly, being able to decide which action to perform autonomously, without central replanning, programming or human control [18].

Latest advances towards added value manufacturing were traduced in recent years mainly by the *Industrie 4.0* initiative or also known as the 4th Industrial Revolution [19]. The evolution of industry and its revolutions is represented in Figure 2.1. *Industrie 4.0* is characterized by the fusion of several technologies, like Internet of Things (IoT), Big Data, Cloud Computing and Wireless Sensor Network (WSN), which are emerging in the industrial world recently.

Industry 4.0 is based on the idea of converging both real and virtual worlds, by connecting every physical object to each other, in order to identify themselves to other devices and be able to communicate with each other. Also, based on Artificial Intelligent (AI) principles, it is emphasized the creation of intelligent machines, which make decisions and react to changes like human operators did. Wireless communication technologies together with ubiquitous Internet access provide increasing interaction between machines, devices, sensors and people. Interconnection of objects and people in industrial facilities is the basis of joint collaboration - human-human, human-machine, and machine-machine, in order to reach common goals when performing tasks. All these ideas pave the way for the Smart Manufacturing concept, which is being achieved with the fusion of Cyber-Physical Systems (CPS), Service-Oriented Architectures (SOA), Cloud and Fog Computing into the CPPS concept.

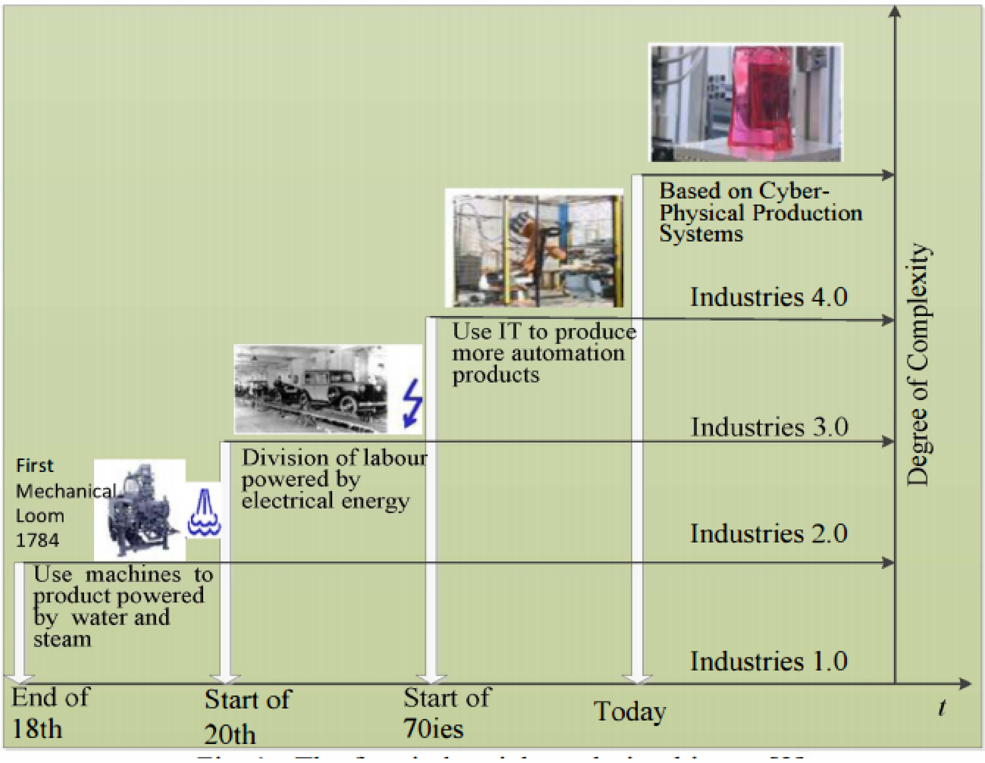

Based on Cyber-Physical Production Systems
Use IT to produce more automation products
Division of labour powered by electrical energy
First Mechanical Loom 1784
Use machines to product powered by water and steam
Industries 4.0
Industries 3.0
Industries 2.0
Industries 1.0
Degree of Complexity
End of 18th
Start of 20th
Start of 70ies
Today
t

Figure 2.1: The four industrial revolutions [1]

2.2.1 Cyber-Physical Systems

CPS are automated systems that allow a complete integration with the ongoing operations of the physical reality, using collaborative computational components and communication, delivering and using SOA for data-accessing and data-processing [12]. CPS rely on the virtualization of all entities involved in the process into an agent-based form. This virtualization consists, in this work, in the creation of Smart Components [20, 21], which are basically digital twins (realistic digital copies of industrial equipment and manufacturing processes [22]) or logical representations of physical devices (like industrial equipment). These Smart Components present sensor data preprocessing methods and can be implemented inside a physical device, with the required computing power, storage, and network connection. These devices are known as Fog Nodes. By using this digital copies, the user gains flexibility in the design and operation of the machine and manufacturing process, reducing cycle times, improving efficiency, and increasing market agility [22]. In the CPS, sensor information is collected from the real world by the sensor equipment and sent afterward to the Smart Component, where computation modules process the measured data. After processing the data locally, the Smart Component sends the information to a Cloud infrastructure, for a high level of data analytics and autonomous decision making. Decisions are sent back to the Smart Components in the form of control commands to modify the surrounding physical environment.

There is no standardized architecture for a CPS, but based on its definition, involved concepts, and objectives, Han *et al.* [23] proposed a division into four essential layers, also represented in Figure 2.2:

- Perception layer, where all of the system's physical devices are located, allowing the monitoring of the process and subsequent actuation, if necessary.
- Network layer, where all the data obtained in the previous layer is transmitted. In this layer, several networks can be used, like Bluetooth, Wifi, Zigbee, etc.
- The Middleware Layer is used for information processing and it can be divided into two sub-layers: the low-level layer, characterized by the Fog Nodes performing the low-level data processing; and the high-level layer, characterized by the Cloud performing the high-level data processing. Not all CPS have this layer divided into two, some only include the high-level Middleware layer.
- The Application Layer is where the data obtained can be monitored by the system end-user through smart applications and HMI solutions.

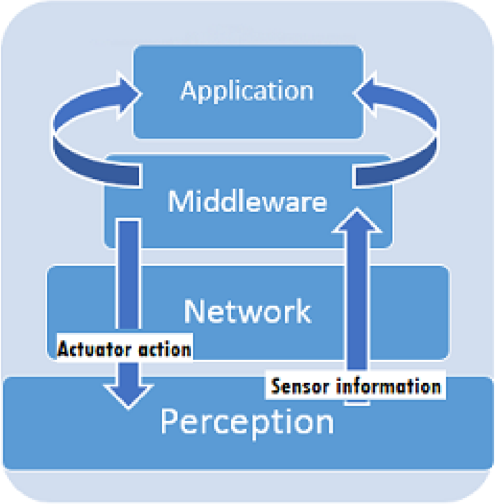


Figure 2.2: CPS Layers

As represented in Figure 2.2, the resulting sensor information from the monitoring of the process and physical devices in the Perception layer is transmitted through the Network layer to the Fog Nodes (low-level Middleware layer), in order to process the sensor data at a low level. This data is sent afterward to the high-level Middleware layer, once more through the Network layer, where the high-level data processing takes place, resulting in adaptive decision making, which is sent back to the system's actuators in the perception layer in the form of actuation commands. All the information obtained about the process can be seen in the Application layer.

One of the main aspects of a CPS is the ability to control and monitor system devices remotely through a network connection. With the implementation of IoT, those devices are able to swap information through Internet. This process involves several phases [24], which are illustrated in the CPS architecture shown in figure 2.3.

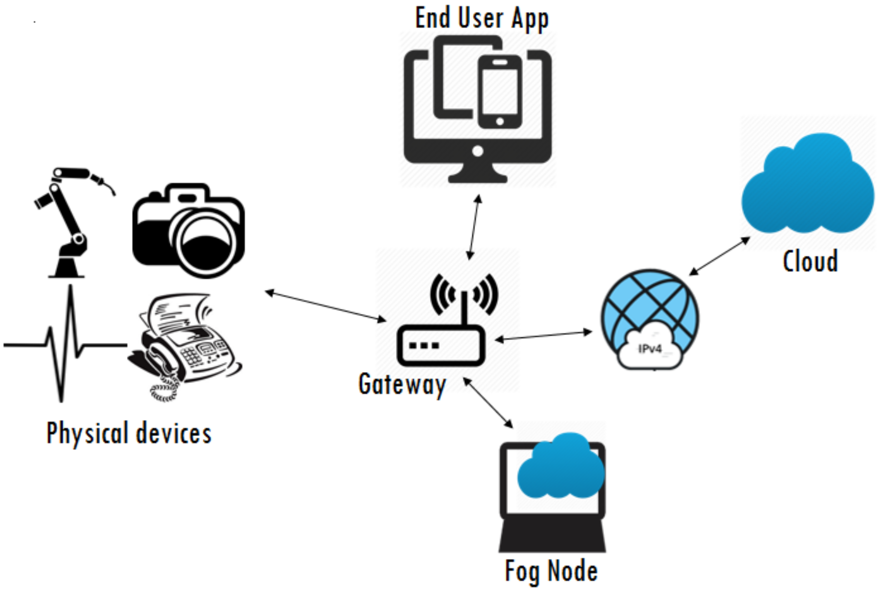


Figure 2.3: CPS Architecture

First, the CPS acquires sensory data from sensors embedded in the physical devices (monitoring phase), which is transmitted through a gateway (networking phase) to an external device, such as Cloud or a Fog Node, where it can be processed and examined remotely, as long as there is a network connection (computation phase). The Fog Node is used to compensate the limitations of the Cloud in terms of providing services that required local access to physical devices. Also, the actions to be sent to the actuators can be decided on the computation phase or directly sent by the end user (actuation phase). After the processing and actuation phases, the information can be displayed in an end user application (displaying phase). These stages are sequential in a cyclical process since the monitoring phase will collect data that traduces the impacts in the physical process of the previous actuation phase.

2.2.2 Cyber-Physical Production Systems

One of the most promising fields where a CPS can bring great improvements is in the manufacturing environment, expanding its resilience and adaptability. Given this, the notion of Cyber-Physical Production System (CPPS) was created, as a specialized concept of a CPS [25].

A CPPS, like a CPS, consists of independent elements in cooperation that communicate across all levels of production, from manufacturing processes in the industrial machines to the logistics networks. In addition to typical CPS modules, the CPPS requires specific characteristics, such as flexibility in the production system and adaptability of the assembly tasks. It enables communication between humans, machines, and products, being able to acquire and process data, and interact with humans via interfaces [12]. CPPS differ from the traditional automation solutions since they have a more decentralized way of functioning on higher levels of the hierarchy, although the control and field levels still exist, including the PLCs, in order to provide good performance in control loops.

For CPPS, a 5-level architecture was proposed by Lee, Bagheri, and Kao [26], called 5c architecture, represented in Figure 2.4. This architecture serves as a guideline for the creation of a CPPS, from the initial data acquisition to analytics, and finally achieving the wanted production system.

The 5c architecture is composed by the following levels:

1. Smart connection level: This level is where the data from the physical devices is acquired. This data has to be reliable and precise, and it can be obtained, not only from sensors but also from enterprise manufacturing systems. Since the data is acquired from several different devices, it is important for the acquisition and transfer methods to be based on non-contact remote techniques, allowing a simple data transmission to the central server. Besides this, the selection of the sensors used (type and specifications) should be adequate to its purpose, being an equally important step in this level.

2. Data-to-information conversion level: This level consists in deducing relevant and meaningful information from the data acquired, effectively correlating and computing massive

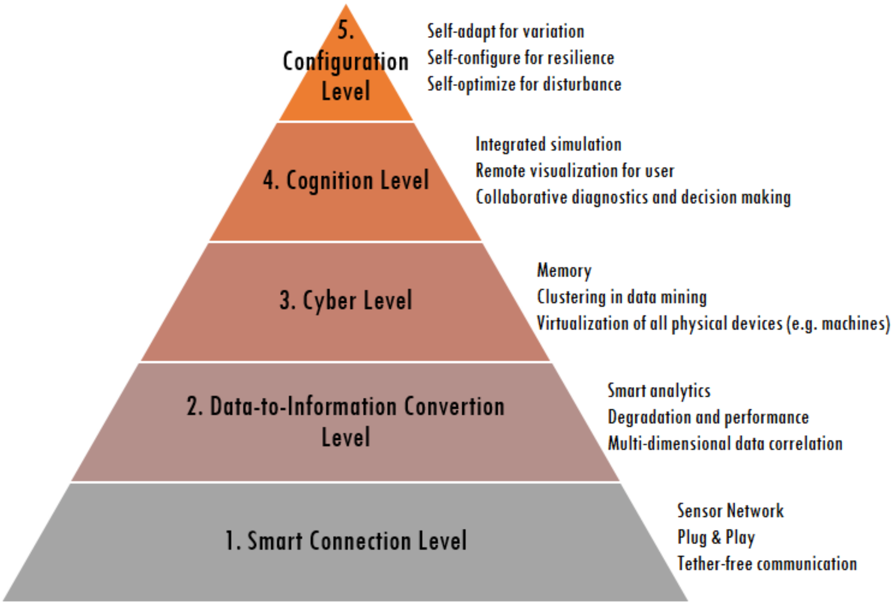

5. Configuration Level
Self-adapt for variation
Self-configure for resilience
Self-optimize for disturbance
4. Cognition Level
Integrated simulation
Remote visualization for user
Collaborative diagnostics and decision making
3. Cyber Level
Memory
Clustering in data mining
Virtualization of all physical devices (e.g. machines)
2. Data-to-Information Convertion Level
Smart analytics
Degradation and performance
Multi-dimensional data correlation
1. Smart Connection Level
Sensor Network
Plug & Play
Tether-free communication

Figure 2.4: 5C architecture for CPPS

and multi-sensor data. Several methods generally utilized for data-to-information conversion are used. These methods are used in several areas but are most applied in prognostic and health management. Through significant information acquisition, like remaining useful life of the machines, or even physiologic information of the manufacturing process workers, this level of the CPPS creates self-awareness and environment awareness to the machines.

3. Cyber level: This level acts as the information hub, where all the information from every physical device in the production system is gathered, forming a network of physical devices. Given the huge amount of data gathered, specialized analytic methods need to be applied in order to obtain additional information that gives some insight on particular machines on a production line. This allows a machine to compare itself with other machines in matters of performance.

4. Cognition level: This level is where the knowledge of the production system being monitored is presented to the users, being of extreme importance since it leads to some decision taking with regard to the production system. Using the scenery where the information available is related to the machine's status and comparison between them, the user can observe this information and use it, for example, to prioritize tasks. The use of proper info-graphics is essential in this level, as it permits the complete transfer of acquired knowledge to the users.

5. Configuration level: The configuration level provides the feedback from the cyber system to the physical system, also supervising and controlling the machines so that they can self-adapt. This step applies to the monitored system the decisions made in the cognition level.

It can be concluded that the implementation of a CPPS in a factory is extremely beneficial, providing several advantages. Developing a Smart Component for each factory component, a historical data record can be captured and converted to meaningful information, allowing the creation of steps to permit self-awareness. Also, with the knowledge that the Smart Component empowers to the physical devices, the factory can become self-configurable and self-maintained, which reduces costs and improves productivity, as it assures near zero downtime production and allows a close to optimal production planning and inventory management.

2.3 Self-Adaptive Systems

After acquiring the information that allows to perceive the human operator's performance, the CPPS should be capable of self-adapt to changes in the production process and the operator's indicators in real time, in order to manage and control actuation devices and the global behavior of the CPPS in accordance with several performance indicators, with little or no human intervention. Modern systems usually operate in dynamic environments, dealing with highly changing operational conditions, which require self-adaptation capabilities. This has been generally recognized as an effective solution to deal with the increasing unpredictability, complexity, and dynamic nature of these systems.

2.3.1 MAPE-k

A well-recognized engineering approach to realize self-adaptation is by means of a feedback control loop called Monitor-Analyze-Plan-Execute plus Knowledge (MAPE-K) feedback loop. The MAPE-k control loop was first introduced by IBM in their Autonomic Computing White Paper [27], where the vision of autonomic computing was explained as the ability of a computational environment to dynamically adapt to changes according to business objectives, as well as to manage itself. Figure 2.5 shows a self-adaptive system, extended with a feedback loop composed of the following MAPE-K components [2]:

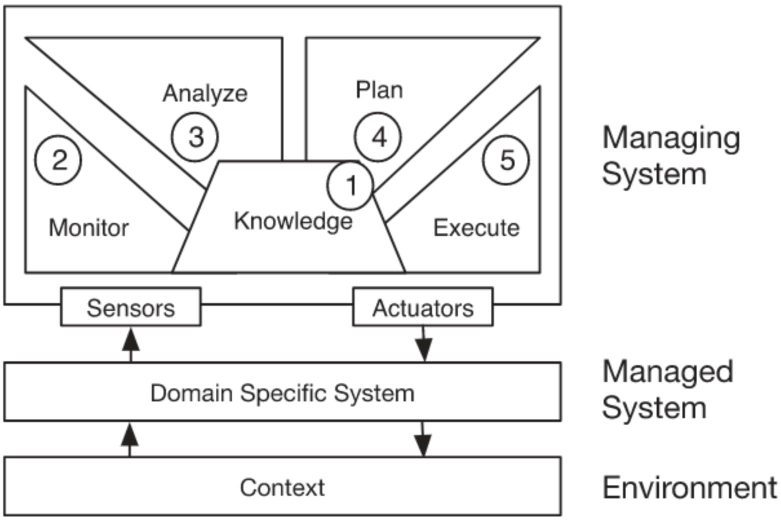


Figure 2.5: MAPE-K model for self-adaptive systems [2]

1. Knowledge component: holds data of the managed system and its environment, as well as adaptation goals and other relevant states that are shared by the MAPE components. It is created by the monitoring component but updated by the execute component.

2. Monitoring component: acquires data from the managed system and its environment through sensors, saving data to the Knowledge component accordingly. This component can aggregate and filter these data until it determines a symptom that needs to be analyzed.

3. Analyze component: uses knowledge to perform data analysis and determine if there is a need for adaptation of the system with respect to the adaptation goals. If there is a need for adaptation, the Plan component is triggered by logically passing a change request.

4. Plan component: this component acts if adjustments to the environment are required, composing single commands or complete work flows with adaptation actions fundamental to accomplish the system's objectives.

5. Execute component: the actions needed for adaptation are executed by this component through effectors or actuators, leading to the wanted adaptation. The actions are the ones recommended by the plan component.

All of these MAPE components can communicate directly or indirectly by sharing information in the knowledge repository.

2.4 Scope of Proposed Approach

Considering the CHMI problem definition, mainly in the HMC spectrum of proposed solutions, it can be concluded that there is a lack of advances for collaboration between the human and machine components of the production process. Also, given the CPS importance in the present reality of industrial environments, this dissertation focus in the study of the cooperative relationship between industrial robots and shop-floor operators, by adapting the robot task execution to the operator's working conditions, implementing a CPPS and the MAPE-k model for self-adaptation. To ensure the system's reliability and help perceive how the collaborative process can help improve the operator's work circumstances while in interaction with industrial machines, the following use cases were defined:

- Robot adaptation to the operator's psychological state: this point consists of six tests to cover all possible classification combinations of the inferred psychological states. These combinations are induced on the tested individual, checking if the machine perceives them, adapting its actions in order to improve the worker's state.
- Adaptation to the operator's task order preference: in a process composed by a set of tasks, those tasks are executed alternately by the operator and the machine. This is tested by changing the task order performed by the operator, evaluating if the machine automatically adapts.

These two use cases involve a continuous monitoring of the operator during the realization of the manufacturing process. This is achieved by using additional external sensors that allow inferring the user's state, namely his stress and fatigue levels, and task order preference. The machine will be able to operate a set of pre-defined but configurable tasks autonomously, where these tasks will be alternately executed with another set of tasks that are performed by the operator. Once the machine's task is completed, the operator is required to intervene by carrying out a task himself, that will lead to the execution of another task of the machine. This sequence of tasks continues until the overall process is finished. According to the technology readiness levels proposed by the Partnership for Robotics in Europe [8], this implementation classifies in a Level 5 of Human-Robot Interaction and a Level 2 in Task Adaptability.

Chapter 3

Human Operator Monitoring

In a cooperative human-machine industrial environment it is essential for both parts to mutually adapt, as it allows the process to become more productive, secure and comfortable to the human worker. As the human is required to adapt itself to the machine's performance, the opposite is also true. In order to enable machine adaptation, the machine should be able of perceiving how the process is being executed by its human co-worker, or how it is affecting him. Assessing, tracking and measuring the operator's performance over time, several metrics are often used, which allow perceiving when and how the process or human operator decreases or increases their efficiency.

In this chapter, the monitoring of the human operator is addressed, namely in terms of stress and fatigue levels, as well as their task order preference throughout the production process. The effect of stress and fatigue in biosignals is overviewed, along with a literature review of computational methods to infer those states and wireless sensing equipment, such as wearables, that can detect stress and/or fatigue.

3.1 Stress & Fatigue

Generally, high levels of stress and fatigue are directly related to poor human performance, leading to low productivity and even safety and reliability issues. With this in mind, stress and fatigue levels can be good key performance indicators (KPI), allowing to gauge the effectiveness and quality of the operator's work. Normally, people are poor self-assessors of how fatigued and stressed they are throughout the manufacturing process, so measuring these KPIs is of overriding importance to the reduction of risk situations and also optimize the productivity by improving the operator's work conditions.

Stress can be defined as the conscience of not being able to deal with the demands of a certain work environment. It is associated with a negative emotional response, which leads to serious concern from the affected person [28]. Immediate stress threats provoke the body's "fight or flight" response, derived from stress stimuli. This response is based on the body's hormone secretion, which causes many physical changes, approached later in Subsection 3.1.1. On the other hand, fatigue is defined as a temporary inability, or the decrease of ability, to answer to a certain situation. This occurs mainly due to activity overloading, where this activity can be from emotional, physic or mental nature. It provokes changes in the psychological and physiological state of an individual during and following prolonged periods of demanding activity, which impairs their work performance and puts them at increased risk for accidents and health problems [29]. It is important to note that the human response to stress can be produced until he is feeling fatigued or exhausted. Given this, fatigue can sometimes be considered an outcome of stress.

Fatigue and stress are psychological conditions that can damage severely human health and working performance. In the American industry, for instance, fatigue in the workplace costs slightly 77 billion dollars per year and is considered a major factor in the stress levels of several employees. According to the researchers, the rate of lost productivity was much higher for workers with fatigue (66%) when compared with 26% for workers without fatigue, while 35% of workers say their jobs are harming their physical or emotional health [30, 31]. Facing this, it is essential to continuously monitor operators' stress and fatigue, this can be achieved by first understanding the effects of these states in the human body, in order to define novel approaches of obtaining stress and fatigue metrics.

3.1.1 Electrocardiogram & Photoplethysmograph

The human Autonomous Nervous System (ANS) controls the unintentional actions of the body, regulating the glandular activity, smooth musculature and cardiac musculature. This system is subsequently divided in the Sympathetic Nervous System (SNS) and Parasympathetic Nervous System (PNS), both responsible for the cardiac activity. SNS is responsible for the control of emergency situations, while SNP is responsible for the relaxing activities and the maintenance of the body. In normal situations, there is a balance between the activities of SNS and SNP: if the activity of the SNS increases, the heart rate will also increase, while with an increase in SNP's activity, the heart rate will return to normal [32]. However, when the human mind perceives an event that causes stress, this balance is broken, releasing adrenaline and cortisol, which consequently increases heart rate and contracts blood vessels [33].

Additionally, through inter-beat intervals (IBIs), i.e. the time in milliseconds in-between two subsequent "R" waves of an Electrocardiogram (ECG), the Heart Rate Variability (HRV) can be obtained, which can be used to realize if a person is under a stress situation. HRV can be determined in both frequency and time signal domain, being that due to the amount of data necessary for an accurate analysis in the time domain, is preferable to use the frequency domain. SNP and SNS's activities are associated with two frequency bands of HRV. The high-frequency component (0.15 to 0.4 Hz) measures the influence of SNP's activity, while the low-frequency component (0.04 to 0.15 Hz) provides an index for the effects of SNS in the heart [32]. There is an inverse relation between stress and the normalized high-frequency component from HRV, so it's possible to determine stress levels based on the HRV.

As it occurs on stress detection, ECG and HRV are revealed as excellent indicators of fatigue existence on an individual. The heart rate substantially lowers when signs of fatigue are detected and great changes happen in HRV due to the diminishing of alert state [34]. Zhao *et al.* [29] presented results in an assessment of mental fatigue in a driving simulator, where they show that fatigue leads to a decreased trend of the low frequency of HRV and an increasing trend at the high-frequency component. Based on this, one can conclude that there are several methods to detect situations of stress and fatigue using the ECG signal, while the most commonly used method is the HRV frequency domain analysis [32, 35].

Using a Photoplethysmograph (PPG) valuable information about the cardiovascular system is provided. According to Yoo & Lee [36], the Pulse Rate Variability (PRV) extracted from PPG can be a potential replacement of HRV, so, when detecting mental stress, there are cases where several signals are used for better results, for instance, using the pupil's diameter, ECG and PPG, where the data obtained is analyzed through methods of soft computing [37].

3.1.2 Electrodermal Activity

One special feature of the SNS is that it exclusively controls the sweat glands and the skin's blood vessels. Thereby, the skin's conductance, which is directly related to the number of active sweat glands, is a reliable way to detect when a stressful situation occurs. Electrodermal Activity (EDA) is registered by measuring the skin's conductivity, given that the relation between that conductivity and sweat secretion is proportional. The Conductance Level (SCL) is normally measured on the palms of the hands or feet, where the sweat glands density is higher [3]. The SCL translates a measure of psychophysiology activation, which is dependent on the characteristics of the individual.

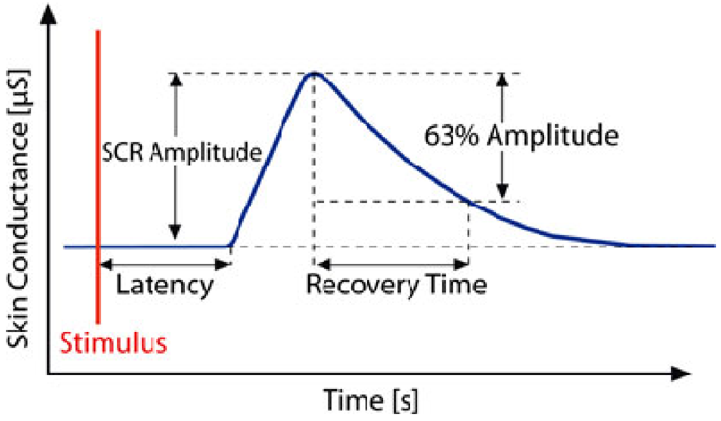


Figure 3.1: Skin Conductance Response [3]

The peaks in the signal (Figure 3.1) called Skin Conductance Responses (SCRs), occur in reaction to single stimuli such as warning events and appear between 1.5 to 6.5 seconds after the stimuli. The SCR can be described through its amplitude, latency and recovery time. Xia *et al.* [38] proved that, when in a state of mental fatigue or stress, the SCL increases, as well as the SCR. Setz *et al.* [3] was able to detect stress situations with a maximum precision of 82.8%, by monitoring EDA with a glove like a device. They used a threshold to detect the EDA peaks, and consequently infer stress. Hernandez *et al.* used EDA measurements to detect stress levels in a Call Center, where EDA measurements were crossed with the workers' opinions, as they considered being under a stress situation, obtaining a maximum precision of 78.4%. There were also other cases where the SCR's utilization is combined with other signals, like the ECG, using also the HR and HRV to detect situations of stress. To do so, data is obtained from the tested people in stressful situations in order to train and validate the detection system [39]. Healey and Picard performed stress detection in drivers, using ECG and respiration monitoring, as well as other signals, defining three distinct situations to differentiate three stress levels, namely resting situation, driving in a highway situation and driving in the city. The biggest correlation with the driver's stress level was obtained through the heartbeat and skin conductance, obtaining a precision of 97% [40].

3.1.3 Electroencephalography

A state of mental fatigue leads to a reduction in the cerebral activity and therefore drowsiness and exhaustion. To better understand the occurrence of fatigue in a human, it is possible to quantify the signals of an Electroencephalography (EEG) in terms of frequency band. This enables the access to the phase in where the human is in between the state of being awake and asleep (stage one of sleep), in order to isolate the relevant signals to identify fatigue.

From all of the existent cerebral waves, the ones that directly reveal the level of fatigue are the Delta and Theta waves. Delta waves are characterized to be slow waves between 0.5 and 4 Hz, which are presented during the transition to the drowsiness state and during sleep. Theta waves are in the frequency band from 4 to 7Hz and are associated with low levels during the sleep stage and tend to increase in cases of fatigue. On the other hand, studies show that the Beta Waves (between 13 and 30 Hz) translate the increase of alert and excitement states, being possible to admit that with the reduction of these waves' activity, one might be in a fatigue state [34, 41]. Also, Simon *et al.* study fatigue effects on drivers considering Alpha waves (7 Hz to 13 Hz), where they demonstrated that Alpha waves are directly related to the assessment of changes in visual perception performance and can be used to detect early stages of fatigue.

3.1.4 Electromyography

The Electromyography (EMG) is used to measure the muscular activity through the detection of superficial tension, which occurs when a muscle is contracted. Therefore, given that stress leads to disorders in the musculature, namely in upper muscular tension during both physical effort and resting, is possible to detect the occurrence of a stress situation through an EMG [42]. EMG has been used to study emotional stress in several situations, most of them using facial muscles [32, 43]. However, there are also cases where other muscles are used, for instance, Rissén *et al.* [44] studied the presence of stress in thirty-two women in repetitive jobs, as well as its influence in the muscular activity of the trapezius, concluding that exists a substantial correlation between the two.

3.1.5 Motion

The stress and fatigue levels are generally obtained by measuring biometric signals. However, they can also be measured using human movement patterns, through either vision techniques or movement sensors. There are some cases of fatigue detection by filming the individual's face while under test. Vision techniques were used to evaluate the skin tone, closing eyelids, yawning, eye movement and other indicators, knowing that in states of fatigue the act of blinking and yawning happens more often and ocular movement is reduced. Ji *et al.* [45] detected fatigue through vision, by developing a probabilistic model to predict fatigue of a human driver based on visual clues as eyelids, ocular and head movement, as well as facial expression. Also, Matsushita *et al.* [46] used accelerometers to detect movement and developed a fatigue detection system, taking a shape of a helmet that detects head tilting.

It is important to refer that in most of the stress and fatigue detection systems, accelerometers are used to complement biometric signals and not as main sensors. This fact is essentially due to the need of taking into consideration the changes that physical activity provokes on the biometric signals. For instance, the heart rate increases significantly when changes in posture or physical activity happen. Facing this, sometimes is possible to confuse alterations due to stress and fatigue with effects derived from physical movement. Physical activity also affects skin conductance levels and provokes breathing arrhythmias. Sun *et al.* [47] used an ECG, combined with EDA measures, and an accelerometer, in order to differentiate the effects of physical activity from stress. Data from 20 participants were collected in three different activities. Better precision in stress classification was obtained using the data from all sensors and the Decision Tree J48 method, obtaining a 92.4% precision.

3.1.6 Stress & Fatigue Classification

The physiological metrics described in the previous subsections are not, by themselves, reliable indicators to determine if an individual is under stress and/or fatigue. However, when combined, they can provide a complete stress and fatigue inferring system. To accomplish this, it's essential to use machine learning techniques regarding sensor fusion and classification methods. Consequently, sensor fusion will improve the process of stress and fatigue classification by automatizing the metrics analysis. The computational models will use a set of combined stress and fatigue metrics as inputs in order to obtain the levels of stress and fatigue at the time when those metrics were measured [48].

A survey was performed, in order to identify some of the most used computational methods for stress and fatigue detection. This subsection briefly reviews the basics of those techniques, where they were used, and their reported accuracy. Also, a resume of the usage of the considered classification techniques, considering the accuracy and input signals for each one, is represented in Table 3.1. To obtain this Table a research was performed with the terms "stress classification" and "fatigue classification" from which a total of 4 280 000 results were obtained for stress and 1 920 000 for fatigue. From those results, only the ones accessing human fatigue or stress were considered, then removing the one without experiments or with no use of body signals.

3.1.6.1 Bayesian Classification

Bayesian classifiers are based on the application of Bayesian analysis to classification problems. If a record of samples is denoted as t, and those records have been assigned to q known class labels, there is a class label set $C = C_i$ with the greatest posterior probability for this record that will classify it. This probability of a record t towards class label C_i is given by [59] and represented in Equation 3.1.

$$P(C_j|t) > P(C_k|t), j = 1,...,q; k = 1,...,q; j \neq k \tag{3.1}$$

Table 3.1: Review of techniques used for modeling stress and fatigue

Modeling Technique	Accuracy	Inputs for model
Naive Bayesian classifier	78,65% [49] [50]	BVP, GSR, PD and ST
Bayesian Networks	86% [47] Not Revealed [51]	ECG and GSR Facial Expression, Eye/Head Movement
Neura Networks	90% [41]	ECG
Decision Trees	88.02% [50] [49] 92.4% [47] Not Revealed [52] Not Revealed [53]	BVP, GSR, PD and ST ECG, GSR and ACC Pedal controlling pattern EEG
Support Vector Machine	90.10% [50] [49] 84% [47] 78.5% [37] 87.2% [54]	BVP, GSR, PD and ST ECG, GSR and ACC PD, ECG and PPG EEG EOG
Kth-Nearest Neighbor	88.89% [55] 88.28% [56]	EEG ECG
Fuzzy	Not Revealed [57] [35] [32] 100% [58] 99.5% [39]	ECG Eyes/Mouth Movement GSR and ECG

From Bayes rules, the posterior probability of a record t towards C_j is given in Equation 3.2, where P(t|C_j) is the conditional probability of a record t for a class C_j.

$$P(C_j|t) = \frac{P(t|C_j)P(C_j)}{P(t)} \tag{3.2}$$

There are several Bayes Classifiers, for instance, the Naive Bayes classifier makes use of all the classes contained in the record sample, and analyses them individually and independently. It considers each attribute separately, working under the assumption that one class works independently of the other attributes contained in the sample. Naive Bayesian classifiers have been used to classify if the individual is under stress or not, in several stress detection systems [49, 50]. Also, Bayesian Networks (BN) were used to detect if the stress measures were high or in a baseline, being that only the Dynamic BN used by Liao *et al.* allowed to show how the properties of stress modified overtime [51, 47].

3.1.6.2 Artificial Neural Networks

Artificial Neural Networks (ANNs) are based on neural connections, capable of learning and reacting. Figure 3.2 represents an ANN diagram example. ANNs use basis functions, which are nonlinear functions of a linear combination of inputs, where the coefficients in the linear combination work as adaptive parameters [60].

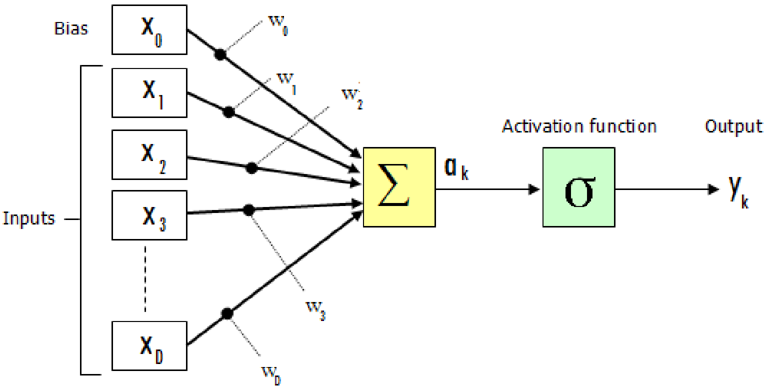


Figure 3.2: Neural Network Diagram

A ANN model can be characterized as a succession of functional transformations, where M linear combinations of D inputs (x_1,...,x_D) are constructed, as shown in Equation 3.3, where j=1,...,M and (1) indicates the layer of the network, in this case the first one. $w_{ji}^{(1)}$ are weights and $w_{j0}^{(1)}$ biases (parameter that allows fixed off sets in the data).

$$a_j = \sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)} \tag{3.3}$$

The parameter a_j is known as activation and is transformed by an activation function that is chosen accordingly to the nature of the data and the distribution of target variables. For multiple binary classifications, each output unit activation is transformed using a sigmoid function so that the network outputs y_k can be obtained, using the Equation 3.4, where k=1,...,K, being K the total number of outputs.

$$y_k = \sigma(a_k) \tag{3.4}$$

One of the most important parts of implementing an ANN is choosing the number of hidden layers and neurons per layer. A small number of hidden layers and neurons could inhibit the differentiation between complex patterns, while a large number could bring weak results mainly due to lack of ANN convergence and the necessity of a large number of input data points. Nevertheless, Patel *et al.* used an ANN to classify drivers' fatigue were the accuracy for one and two hidden neuron layers were similar [41].

3.1.6.3 Decision Trees

The decision tree classifier is a popular and practical method that uses a "divide-and-conquer" approach based on a tree-like structure. Each node tests an attribute, each branch is an output of the test, and each final node (leaf node) represents classes [48]. It basically involves the disassembly of a complex decision into a merge of various simpler decisions to obtain the desired solution.

The classification of an unknown sample is performed by testing the values of the attributes in each node. Consequently, it creates a path until it reaches a final node, where the instance is classified according to the class assigned to that final node. The branches between the nodes provide possible values that these attributes can have in the observed samples, whereas the terminal node determines the final value of the variable. A path is traced from the root node to a final node that has the class prediction for that sample [49]. A generic Decision tree and its constituent parts are represented in Figure 3.3.

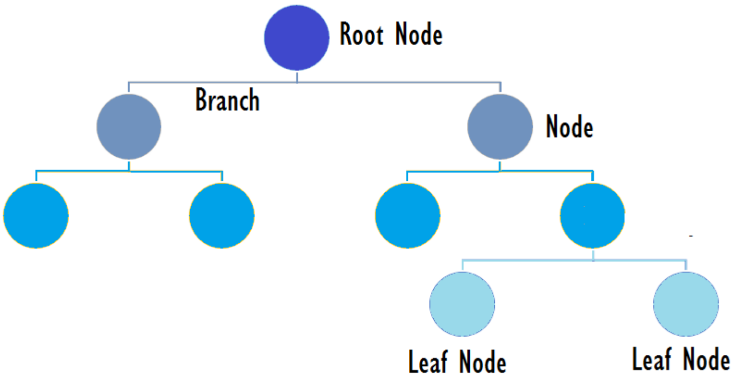


Figure 3.3: Generic Decision Tree model

This method has been used to classify booth stress and fatigue. Zhai and Barreto [50] proposed a system where the combination of Blood Volume Pulse (BVP), Galvanic Skin Response (GSR), Pupil Diameter (PD) and Skin Temperature (ST), by using a J48 decision tree, obtained an accuracy of 88.02%. Sun *et al.* [47] used ECG, GSR, and ACC data to create a metal stress predictor, also using a J48 decision tree as the classifier. To detect mental fatigue in drivers, Kim *et al.* [52] used the driver's pedal controlling pattern, where the sensor data was processed using a combination of Decision Tree learning algorithms, with ID3 algorithms to determine the shape of the decision tree. Also, to detect driver's fatigue, in this case with EEG, Shen *et al.* [53] used a Random Forest, which is an estimator that fits a number of decision tree classifiers on various subsamples of the dataset so that the output is the plurality vote of all these decision-trees.

3.1.6.4 Support Vector Machines

Support Vector Machines (SVM) are computational learning methods that transform a classification problem in such way that allows the application of linear classification techniques to nonlinear data. SVMs belong to the class of Kernel Methods that are formed by a general purpose learning machine and a problem specific kernel function. Since linear machine learning methods can only classify data in a linear space with separable features, the kernel-function can be used to obtain the linear feature space by mapping the training data into a higher dimensional space where the data is linearly separable. These Kernel functions can be of many types, from polynomial to Gaussian (Radial Basis) or Sigmoid.

Having a certain training data set that comprises N input vectors x_1,...x_N with matching target values t_1,...,t_N, with t_n being either -1 or 1, the goal is to separate that set into several individual classes. The SVM approaches this problem through the concept of margin, i.e. the largest distance between the decision boundary and any of the samples. The margin's value is given by the perpendicular distance to the closest point from the data set x_n so, optimizing the parameters w and b, the margin is maximized. Equation 3.5 represents this optimization problem [60].

$$\underset{w,b}{\operatorname{argmax}} \left\{ \frac{1}{||w||} \min_{n} [t_n(w^T \phi(x_n) + b)] \right\} \tag{3.5}$$

Given their characteristics, SVMs have been used to foresee stress using BVP, GSR, PD and ST data [50, 49]. An activity-aware mental stress detection scheme using ECG, GSR, and accelerometer data was proposed by Sun *et al.* [47], by using SVM as a classifier. Also using an SVM, but in this case a fuzzy one, Mokhayeri *et al.* [37] proposed a mental stress detection system, using PD, ECG, and PPG. Shen *et al.* [54] used a Fuzzy SVM to classify fatigue. They propose an EEG-based mental-fatigue monitoring system using a probabilistic-based SVM method obtained an accuracy of 87.2%.

3.1.6.5 kth-Nearest Neighbor

The nonparametric method kth-Nearest Neighbor (kNN), is used for classifying observations into groups, based on a set of quantitative variables. The classifier compares a new sample (testing data) with the baseline data (training data), and finds the k neighborhood in the training data, assigning the class that appears more frequently in the neighborhood of k. The k value is by default 1 but varies, in order to find the match class between training and testing data. The default neighborhood setting is “Euclidean” and “nearest”. To find the object similarity in the k neighborhood, the “Euclidean” distance is used [55], as represented in Equation 3.6.

$$d(X_i,X_j) = \sqrt{\sum(X_i - X_j)^2} \tag{3.6}$$

Sulaiman *et al.* used the combination of EEG power spectrum ratio and Spectral Centroids techniques to extract features for human stress, by implementing a kNN classifier to detect and classify human stress from two cognitive states. Also, Wang *et al.* [56] used a kNN classifier with HRV feature-based transformation algorithms, employing a kernel-based class separability as the selection criterion for feature selection, achieving accuracies between 81.06% and 88.28%.

3.1.6.6 Fuzzy Classification

Fuzzy logic systems or fuzzy inference system (FIS) are based on rules containing three conceptual components: 1) a collection of fuzzy IF–THEN rules - the rule base; 2) a database defining the membership functions used in the fuzzy rules; and 3) an inference mechanism that aggregates these rules in order to obtain the outputs of the system from the inputs. The basic structure of fuzzy logic systems is presented in Figure 3.4.

The If-Then rules are expressed in a way that can be easily interpreted, permitting a simple understanding of the system, being expressed directly by human words, where each word is considered a fuzzy set [61]. All fuzzy sets need to be defined by membership functions before being used to build If-Then rules. In a Fuzzy set, contrary to the classical sets, the membership of elements does not use binary logic, i.e, there isn’t a definition only to whether the element is from a certain crisp set or not. In a fuzzy set, elements have a degree of membership (from 0 to 1) to a certain membership between belonging there and not belonging [61].

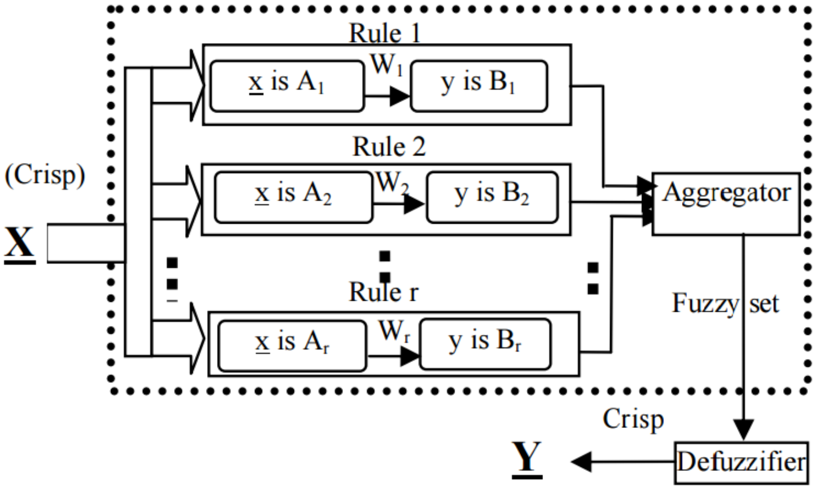


Figure 3.4: Block diagram of a fuzzy inference system [4]

Fuzzy-based techniques are one of the most used computational methods to infer stress and fatigue. Kumar *et al.* [57] used a fuzzy-based modeling technique to filter uncertainties regarding the relation between physiological parameters and the workload score of individuals, evaluated by their HR. Also, Kumar *et al.* [35] proposed a fuzzy clustering and robust identification techniques for mental stress assessment using HRV, since there were several uncertainties due to individual variation. The HRV analysis was also used by Rani *et al.* [32] to infer the stress condition based on the level of activation of the sympathetic and parasympathetic nervous systems, utilizing fuzzy logic in order to allow a robot to recognize human psychological states like stress. Another stress detection system based on fuzzy logic was proposed by de Santos Sierra *et al.* [39], where the behavior of an individual under stress stimuli was described, detecting stress with an accuracy of over 90%, using GSR and HR to supply information on the psychological state of the individual. Fuzzy classification was also used by Azim *et al.* [58] to detect fatigue. A non-intrusive fatigue detection system based on video analysis of drivers, monitoring information of eyes and mouth, that is further passed to Fuzzy Expert System to classify the true state of the driver. This system generated an average accuracy of 100% on the videos tested.

3.2 Sensing Equipment

Sensing equipment can be used to monitor the operator's performance since it is important to observe the emotional state during the manufacturing process, namely stress and fatigue levels. This monitoring can be achieved through several sensors, which can be integrated into wearable solutions designed for this purpose, wireless sensor boards, and other proprietary sensing devices. These devices acquire physical data from the environment and human operator using specific sensors, such as biosignals, human motion, and activity tracking. Later, the collected data is the input of data analytics algorithms, in order to calculate meaningful results, namely stress and fatigue [62]. The sensors used in both the wearables and acquisition boards allow perceiving variations in the individual's emotional state that generate several responses in the human body, like bio-electric impulses and muscle contractions. Therefore, most of the sensors need to be placed on the human body. Other solutions, such as vision systems, are not as intrusive. In situations where the person needs to move freely, the presence of sensing equipment placed in the human body could create an obstacle in the movements, becoming uncomfortable, and even sometimes disconnecting from the body and stopping acquisition process.

3.2.1 Stress Detection Wearables

Nowadays, there are several wearable solutions to monitor stress. While some of the solutions are more reliable than others, many of those solutions take form in wristbands with integrated sensors on them, monitoring a variety of physiological signals such as EDA, BVP, and ST. Computational algorithms translate those biosignals into emotions like happiness, anger, sadness, and stress.

This is the case of the Feel Wristband[1], represented in Figure 3.5. Besides these sensors, the system also provides a mobile app that syncs with the wristband to track the emotions throughout the day, giving the advice to achieve emotional well-being. A team of several people from diverse engineering fields created the Empatica E[2], represented in Figure 3.6, which is a wearable wireless multisensor device for real-time computerized biofeedback. Data acquisition occurs with four different sensors, namely PPG, EDA, 3-axis accelerometer, and temperature. This device operates in

[1] http://www.myfeel.co/gallery

[2] https://www.empatica.com/product-embrace

two modes: 1) streaming mode for real-time data processing using a Bluetooth low energy interface; and 2) recording mode using its internal flash memory [63]. They also created the Embrace Watch that specifically monitors the nervous system with four sensors (ACC, Gyroscope, EDA and Peripheral Temperature Sensor), detecting seizures, patterns of sleep and physical activity.

Figure 3.5: Feel Wristband

Figure 3.6: Empatica Watch

Besides wristbands, there are also other types of wearable solutions, like headsets that detect and monitor mental stress. The InteraXon Muse[3], represented in Figure 3.8 is one of them. This brain sensing headband measures brain signals with seven calibrated EEG sensors and several ACC to detect and measure the activity of the brain, which is then translated into actionable data and sent to a smartphone or computer. Another headset solution is the Emotiv Insight[4], represented in Figure 3.7, which is a wearable EEG headset that can transmit brain wave data wirelessly to a smartphone or computer. This device offers five EEG sensors and two reference sensors providing in-depth information on brain activity. The measurements are based on six key cognitive and emotional metrics: focus, stress, excitement, relaxation, interest, and engagement. These measurements allow an individual to monitor his cognitive health and well being [64].

Figure 3.7: Emotiv Insight Headset

Figure 3.8: InteraXon Muse Headset

[3] http://www.choosemuse.com/

[4] http://www.autodidacts.io/neurotech-hardware-roundup-eeg-bci-tdcs-neurofeedback/

3.2.2 Fatigue Detection Wearables

There are numerous wearable solutions to monitor physical fatigue, most of them dedicated to fitness applications, designed to maximize athletes performance by tracking and reporting strain, sleep, and recovery analysis. That is the case of the WHOOP Strap 2.0[5], represented in Figure 3.9, which collects HR, HRV, ambient temperature and motion via a 3-Axis ACC, allowing the athletes to appropriately balance their strain with the body's recovery, giving daily insights into how fatigued the body is [65].

Besides fitness solutions, there are also wearables dedicated to monitoring fatigue in shop floor workers and drivers. The Fatigue Science ReadiBand[6], represented in Figure 3.10, is a wristband that collects data from wrist movement to calculate the quality and quantity of sleep during the night, as well as activity levels during the day. It is also able to quantify the wearer's effectiveness score, reaction times and relative accident risk.The ReadiBand is approximately 93% accurate at determining when the user is asleep. The fatigue levels are calculated through an algorithm that correlates sleep and fatigue and determines that the user is indeed fatigued if the level reaches the equivalent of having a 0.08 blood alcohol level. This is possible by comparing a person's fatigue levels to the number of errors and delayed reaction time of an impaired person [66].

Figure 3.9: WHOOP Strap 2.0

Figure 3.10: Fatigue Science ReadiBand

The SmartCap system[7], represented in Figure 3.12, is often used by drivers or operators of heavy vehicles. It is a fatigue monitoring tool that provides real-time measurements, based on processed brain wave information. The fatigue information is displayed to the user on a Bluetooth enabled device [67].

[5] https://get.whoop.com/products/whoop-strap
[6] https://www.fatiguescience.com/
[7] http://www.smartcaptech.com/our-product/

Figure 3.11: Maven Co-Pilot

Figure 3.12: SmartCap

Another existent solution is the Maven Co-Pilot[8], represented Figure 3.11, which is an hands-free Bluetooth headset designed to address driver's fatigue and distraction. The device is able to detect the difference between mirror check and moments of micro-sleep and alert the driver in situations of danger [68].

3.2.3 BITalino

From the survey performed regarding wearable devices developed to measure stress and fatigue, no solution was found that allowed to obtain both states at the same time. Also, the very few solutions that could retrieve real time levels of stress or fatigue revealed themselves as extremely expensive devices. So, in order to obtain stress and fatigue values, a cheaper solution would be able to acquire physiological signals, which are processed afterward to calculate stress and fatigue levels. For this purpose, one cheap, mobile and flexible solutions are the wireless BITalino board. This board is a low-cost hardware platform developed by a Portuguese team designed to enable makers to create projects using body signals. It becomes more accessible not only financially but also allowing to attain several psychological and physiological measures in one single device.

BITalino consists of a set of modular blocks capable of measuring physiological signals, connecting with remote gateways over Bluetooth [5]. Each individual block can be physically detached from the main board, allowing the user to apply it in many different ways as represented in Figure 3.13.

The BITalino architecture enables three different configurations:

- Board - is simply the board with no modification, allowing the individual to use the onboard sensors (Figure 3.13 (a));

[8] http://mavenmachines.com/co-pilot/

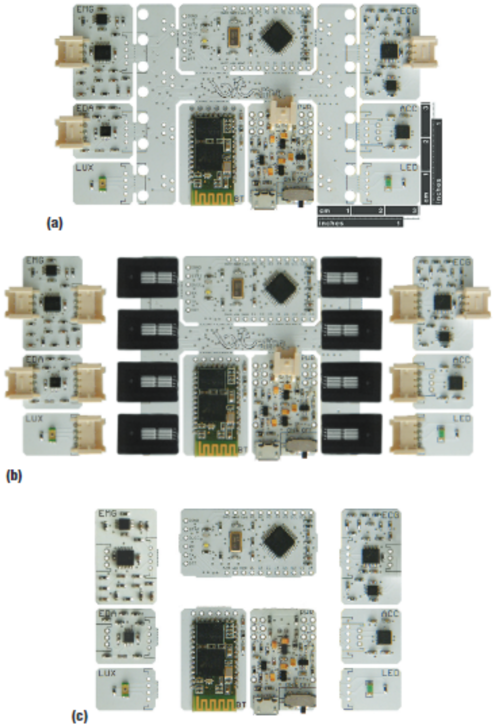


Figure 3.13: BITalino biosignal acquisition hardware in its different configurations: a) board, b) plugged, c) freestyle [5]

- Plugged - plugs are added to the BITalino and only the individual sensor blocks are separated from the main board, allowing people to use different sensor combinations (Figure 3.13 (b));
- Freestyle - all the blocks are detached from the BITalino main board, enabling people to combine them in any way that best suits their projects (Figure 3.13 (c)).

The digital back-end integrates a control block based on a Microcontroller Unit (MCU), a power management block (regulation and battery charger), and a wireless communication block that uses a Bluetooth module [69]. BITalino is battery supported with a 550mAh rechargeable LiPo battery, that in the most battery demanding scenario (all the LEDs are on, and all the sensors are connected and collecting at a sampling rate of 1000 Hz), uses around 65 mA per hour [5], where 60% is used by the Bluetooth module, and 15% by the LEDs. It has approximately 8.46 hours of battery life in full usage, being able to monitor an individual's signals for longer if not all LEDs and sensors are used. All BITalino's primary specifications are described in Table 3.2.

The analog front-end incorporates separate sensor blocks: Electromyography (EMG), Accelerometry (ACC), Electrocardiography (ECG), and Electrodermal Activity (EDA), and also possesses a Light sensor (LUX) and Light-Emitting Diode (LED) block, allowing sampling rates of

Table 3.2: BITalino's specifications

	Specification
MCU	ATMega 328P
Clock	8MHz
Power	Vcc = 3.3V Vss=1.67V;
Battery	550 mAh rechargeable 3.7V LiPo battery
Consumption	∼65 mAh (if at 1000Hz sampling rate and simultaneous acquiring all sensors while all LEDs on)
Data Link	Class II Bluetooth v2.0 module
Analog ports	4 input (10bits) + 2 input (6bits)
Digital ports	4 input (1bit) + 4 output (1bit)
Sensors	EMG, ECG, EDA, ACC and LUX
Actuators	LED
Weight	30g

1, 10, 100 and 1000 Hz. All the sensors are single-ended ($0-3.3$V) and the sensors that measure biosignals (ECG, EMG and EDA) also need an auxiliary mid-supply voltage (Vss =1.65V) as a virtual ground. The specifications to this biometric sensors are summarized in Table 3.3.

Table 3.3: ECG,EMG and EDA sensors specifications

	ECG	EMG	EDA
Principle	Bipolar differential measurement	Bipolar differential measurement	Skin resistance measurement
Number of Electrodes	2 or 3	3	2
Bandwidth	0.5-40Hz	10-400Hz	0-3Hz
Input impedance	100GOhm	100GOhm	-
CMRR	110dB	110dB	-
Range	±1.5 mV	±1.65 mV	0-1 MOhm
Gain	1100	1000	2
Consumption	∼4 mAh	∼4 mAh	∼2 mAh

Besides the board, the BITalino has numerous software components including the firmware on the MCU and programming API's that allow the most development possible. This programming API's are available in several languages including Android, Arduino, C++, Java, LabView, MATLAB, Python, RaspberryPi, and Unity 3D, which allows users to access the sensor data using their own applications. Although the BITalino board is easy to attach and move around, there is the possibility to create wearables from the board, similar to the ones created in projects like an armband controller for the Parrot Jumping Sumo drone using muscle and gestures or Heart rate monitor on bicycle handlebars. Recently the BITalino team made a collaboration with Printoo, a company that produces paper-thin, flexible technology, making both platforms compatible and creating a production version of BITalino that is bendable and appropriate for anatomical forms, facilitating the creation of wearable solutions[9].

3.2.4 Kinect

Kinect v2, represented in Figure 3.14, is a motion-sensing device originally developed for the Xbox One, that contains a 3D depth camera to measure depth in millimeters, an infrared (IR) camera and emitter, an RGB (red-green-blue) camera for color video, and a multi-array microphone [6]. Kinect allows several applications, including tracking humans' position and activities while being performed. Apart from this components, the Kinect device also has a power adapter for external power supply and a USB adapter to connect to the computer.

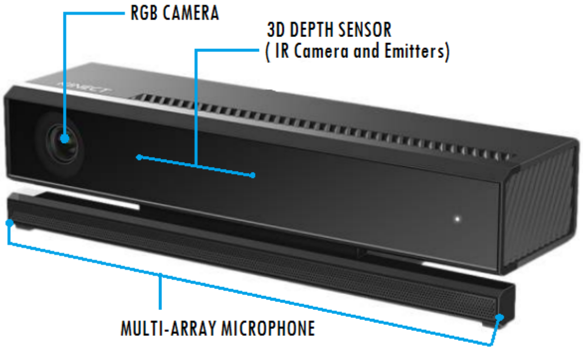


Figure 3.14: Microsoft Kinect v2 and its components

The RGB camera captures the color video data, detecting red, blue, and green colors, and streaming the data by still image frames, supporting a speed of 30 frames per second (fps), decreasing to a minimum of 15fps if in low light [70]. The depth sensor consists of an IR emitter and an IR camera in order to obtain the depth value of every point in the visible area, i.e. the distance from that point to the sensor. The Kinect v2 uses the "Time of Flight" (TOF) method, that determines the depth by measuring the time the light emitted takes from the camera to the measured point and back. Therefore, it constantly emits IR light with modulated waves and finds the shifted phase of the returning light [71]. The developed applications for Kinect, support features like color and images, audio input, and skeletal data, allowing, for instance, to determine the distance between an object and the camera, or recognize and track people using body tracking. All

[9] http://www.bitalino.com/en/community/projects

Table 3.4: Technical specifications of Kinect v2 (based on [7])

Feature	Kinect v2
Depth Sensing Technology	Time of flight
Color Image Resolution	1920x1080
IR Image Resolution	512x424
Depth Sensing Resolution	512x424
Minimum and Maximum Depth Range	50 cm to 4.5 m
Depth Technology	Indirect ToF
Frame Rate	30fps (Color Image: 15fps low light)
Field of View	60 degrees vertical and 70 degrees horizontal
Skeleton Tracking ***(with full skeleton)***	Total 6 skeletons tracked 25 joints per skeleton

of the technical specifications mentioned above, as well as other particularities, are summarized in Table 3.4.

Chapter 4

Implementation

Based on the cooperative human-machine interaction problem defined in Chapter 2, a solution based in a CPPS is proposed. The CPPS is characterized by its self-adaptation skills, which are enabled by the implementation of a MAPE-K feedback loop, that serves as functional architecture. In the context of human-robot collaboration, this self-adaptation consists in the adaptation of the industrial equipment to the human operator, based on the operators' working profile, e.g., preference in a certain order of operation execution while performing a task, and operators' emotional conditions, such as stress and fatigue levels.

In order to achieve this scenario, first, the operator's stress and fatigue monitoring need to be performed, by acquiring and analyzing several biosignals with the BITalino board. Later, collected sensor data is processed in the MATLAB's signal processing tool in order to retrieve metrics and posterior classification of those levels using a fuzzy classifying method. The monitoring of the operator's position and activity tracking is also needed, using for that the Kinect's body tracking function. This function retrieves the area where an operator is executing his tasks, which is one of the inputs for the adaptation of the robotic arm when executing its own tasks in that same area. This chapter will explain all the steps undertaken to achieve the implementation, describing the CHMI architecture and all the methods used.

4.1 Stress & Fatigue Monitoring

Monitoring the operator's stress and fatigue levels required the acquisition of several biosignals in real time, using the hardware platform BITalino. Considering all the APIs that BITalino provides (see Section 3.2.3), the MATLAB API was the one chosen for this dissertation. This API allows the acquisition and visualization of the signals from the BITalino board, needing only to previously define the Bluetooth RemoteID from the BITalino device, the channel from which we want to acquire the signal and the sampling rate (1, 10, 100, 1000Hz). The API was modified in order to allow real time acquisition, obtaining consecutive one-second samples of all the signals. This was possible by using a timer object, which uses callback functions to perform commands every time the timer executes in order to read all the data from the buffer and process the signals.

In this dissertation only the ECG, EDA and EMG were acquired from the BITalino board, all at a 1000Hz sampling rate, as recommended by Němcová *et al.* [72]) in order to preserve more detail valuable for some of the features extracted. For the acquisition of the signals' features, a one minute window of acquired and processed samples was used in order to obtain more reliable values, since some of those features require at least one minute of data. These features are acquired at a low sampling rate of 1Hz since the stress and fatigue variations are very gradual. The following subsection will focus on the acquisition and analysis of this three biosignals captured from the BITalino board.

4.1.1 Electrocardiogram

An ECG is a recording of the heart's electrical activity based on Bipolar differential measurement, where low amplitude bioelectrical signals produced by the sinoatrial node (SA node) trigger heartbeats. These signals can be translated into values through the ECG, allowing to obtain a lot of information, such as health inference and the occurrence of stress and fatigue situations, like mentioned in Chapter 3. The acquisition with BITalino is made with three silver/silver chloride (Ag-AgCl)[1] electrodes connected to 1x3 Electrode Leads[2], represented in Figures 4.1 and 4.2.

The electrodes can be placed in several configurations: 1) in the hands, using only two electrodes one on each hand; 2) in the hands and leg, where it is placed one on each wrist and the ground on the leg; and 3) only in the chest, forming a triangle shape, to the left side of the chest, with the ground electrode as the top corner of the triangle. This last configuration, represented in

[1] https://www.virtuabotix.com/product/bitalino-emgecgekgedaluxacc-sensors-raspberry-pi-pc-android/
[2] https://www.adafruit.com/product/2773

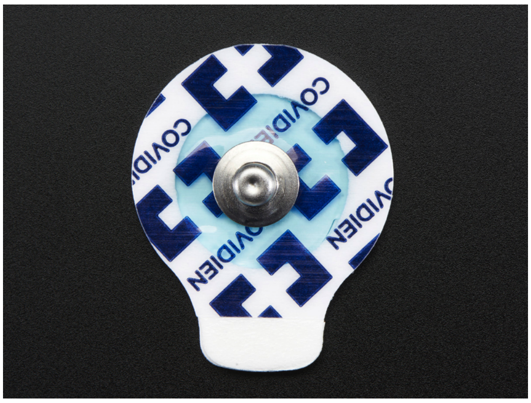


Figure 4.1: Ag-AgCl electrodep

Figure 4.2: 1x3 Electrode Leads

Figure 4.3[3], was the one chosen to acquire the ECG signal, given that facilitates users mobility in comparison to the others.

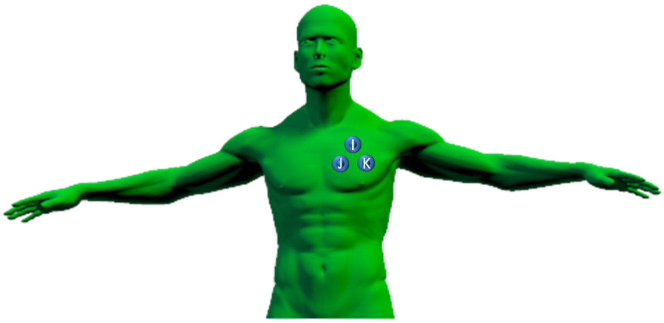


Figure 4.3: Chest electrode placement used for ECG acquisition

4.1.1.1 ECG Preprocessing

After the ECG acquisition, the signal needs to be processed. First, the output values from the ECG sensor in BITalino were converted to standard units of measure using the following Equation 4.1, where ECG_{mV} is the ECG value in millivolts, the ECG_{B} the value obtained from BITalino, Vcc is the operating Voltage (3.3v), n the number of bits (10 bits) and G_{ECG} the ECG Sensor Gain (1100 as showed in Table 3.3). All this values and formula can be seen on the ECG sensor data sheet from BITalino in appendix B.

$$ECG_{mV} = \frac{(ECG_B \times \frac{Vcc}{2^n} - \frac{Vcc}{2})}{G_{ECG}} \times 1000 \tag{4.1}$$

[3] http://www.civicview.org/projects/

Following the data conversion, several filters were applied. First, a Savitzky-Golay filter was used to smooth the signal. This filter is based on local squares polynomial approximation, where first the least square polynomial fit to the signal samples is found, then it was processed through a linear functional transformer to extract the smoothed signal. This filter is equivalent to a digital low pass filter and is used to increase signal-to-noise ratio [73]. After smoothing the signal, it still presented some noise, so to detect the noise's frequency, a power spectral density (PSD) analysis of the signal was done, finding an impulse at 50Hz and some harmonics as well, represented as additional spikes in the PSD at multiples of the fundamental frequency. This noise is commonly considered electrical noise and to remove this electromagnetic interference, a Finite Impulse Response (FIR) stopband filter was implemented to cut off 48Hz to 52Hz noise, i.e., a digital filter with finite-duration impulse response [74]. To implement this filter in MATLAB, the *Fir1* function was used. It applies a Hamming window to design a nth-order, type specified (in this case a stopband) FIR filter with linear phase. The filter's order was calculated by Bellanger's equation (4.2), where fs is the sampling rate, Δf the transition width, i.e., the difference between end of pass band and start of stop band, $\delta 1$ the ripple in passband, i.e., the affordable variation in the original amplitude and $\delta 2$ the suppression in the stop band [75].

$$n \approx \frac{2}{3} log\left(\frac{1}{10 \times \delta 1 \times \delta 2}\right) \times \frac{fs}{\Delta f} \tag{4.2}$$

The effects of the FIR filter, along with the Savitzky-Golay smoothing, in a five-second sample of the ECG signal are represented in Figures 4.5 and 4.4.

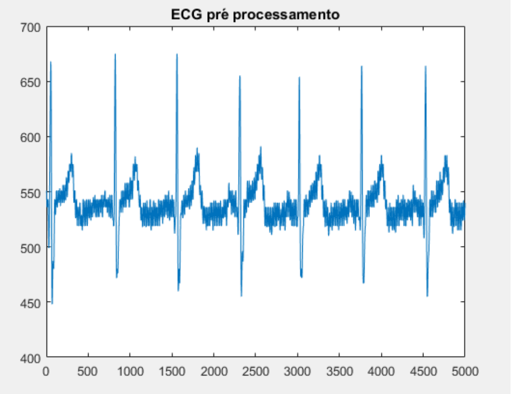


Figure 4.4: ECG signal preprocessing

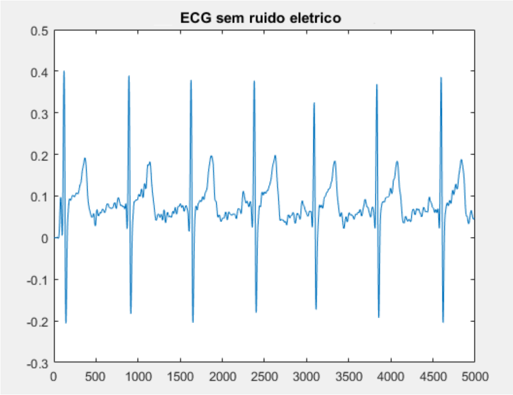


Figure 4.5: ECG signal smoothed and with no electrical noise

The ECG signal can also be affected by muscular noise interference, especially in industrial environments, where the acquisition is done with the operator physically moving around. This noise is very difficult to remove as it is in the same region as the actual ECG signal. However, after trying several filters, the best one to filter this noise was a 6th order lowpass Butterworth filter, at 30Hz cutoff frequency [76], removing some of the muscular interference, with the catch of reducing the amplitude of some of the ECG peak waves. To test this filter, an acquisition was made while slowly moving the arms and consequently contracting the chest area, obtaining the ECG signal showed in Figure 4.6, which after removing the muscular noise became the signal represented in Figure 4.7. As can be seen, with movement, the ECG signal shows strong EMG interference, which is improved after filtering, although not being totally removed.

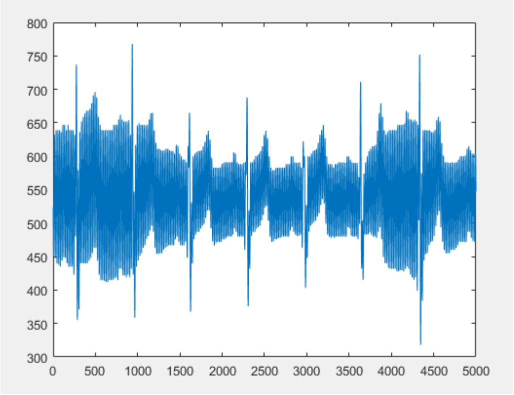


Figure 4.6: ECG signal preprocessed while the moving right arm

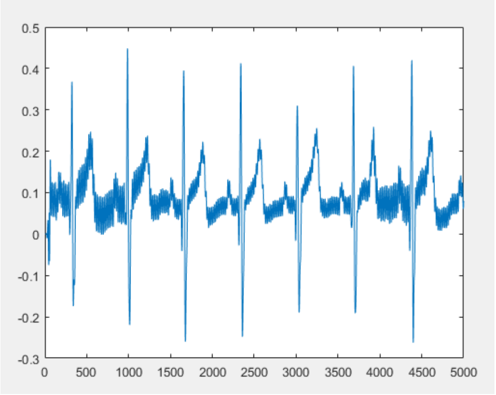


Figure 4.7: ECG signal after filtering, while moving right arm

Another common noise source in an ECG is the baseline wander, a low-frequency high bandwidth component that can be induced by electrodes due to perspiration, respiration and body movements [77]. To remove this baseline wander, a highpass second order Butterworth filter was used to filter lower frequency components, the cut-off frequency was defined by the slowest heart rate (it was considered 35 beats per minute (bpm)), implying the lowest frequency at 0.58Hz (one bpm equals to 1/60Hz). This filter revealed itself not so accurate, as sometimes it did not remove the baseline wander. Figures 4.8 and 4.9 show the best results achieved for this filter application to the ECG signal.

The ECG structure is typically composed of three waves: P, QRS, and T waves, as represented in Figure 4.10. One of the challenges in processing the ECG is the different peaks detection.

After all the noise removed, the ECG signal was squared in order to facilitate this detection.

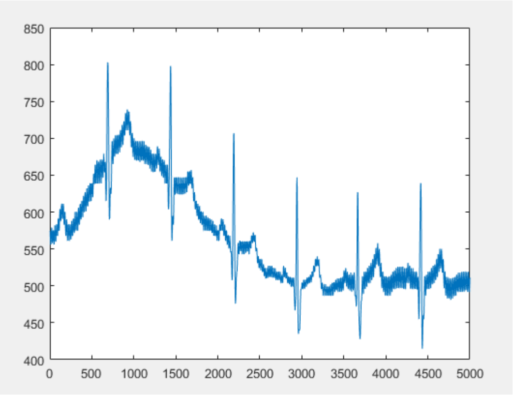


Figure 4.8: ECG signal with baseline wander

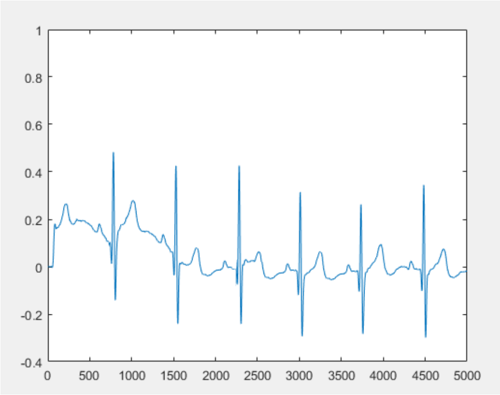


Figure 4.9: ECG signal after filtering the baseline wander

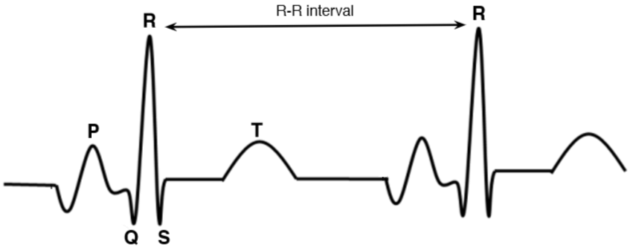


Figure 4.10: P, QRS and T waves

To detect the R peaks was used a MATLAB function, *findpeaks*, that finds local maximum with a defined minimum peak distance of 200ms, taking into consideration that two adjacent R peaks cannot be more close than 200ms at the considered maximum heart rate of 150bpm; and a defined minimum peak height. This minimum peak height is initialized as 0.05, but when the first three R peaks are found, this value turns 0.75 times the mean amplitude value of those three R peaks, in order to more accurately detect the peaks. This function detects where these peaks are and what is their amplitude value for each window of one minute. You can see detected R peaks in Figure 4.11.

4.1.1.2 ECG Features Extraction

As mentioned in Section 3.1.1, the ECG delivers several features that allow detecting stress and fatigue moments. For this dissertation, the features used were the Heart Rate (HR), the Inter-Beat Interval (IBI), and the Low/High-Frequency Ratio (RATIO). Their meaning and abbreviation used

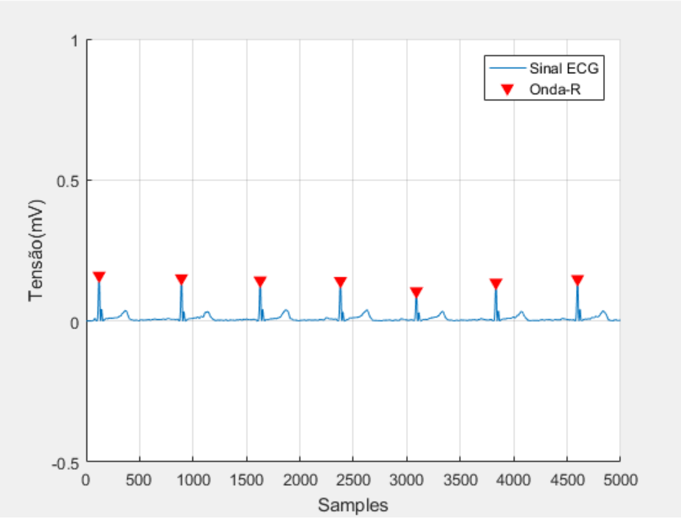


Figure 4.11: Detection of R peaks

can be reviewed in Table 4.1. For the IBI calculation, the R peak detection was used, calculating the time between two subsequent R-peaks in seconds (R-R interval, Figure 4.10). The HR in beats per minute is then easily obtained using the median of RR intervals in seconds and dividing 60 by it. In order to achieve the Frequency Ratio, a Frequency Domain Analysis of the Heart Rate Variation (HRV) is necessary. The HRV is the variation of the intervals between beats, in time, i.e, the variation in the R-R intervals or IBI. The idea for the frequency domain analysis is to transform the HRV from time to power spectral density (PSD), displaying the data in frequency, with two bands of interest: the high frequency (HF) band (defined between 0.15 Hz and 0.7 Hz) and the low frequency (LF) band (defined between 0.04 Hz and 0.15 Hz).

Table 4.1: Features extracted from the ECG signal

Feature	Abbreviation	Meaning
Heart Rate	HR	Heart Rate in beats per minute
Inter-Beat Interval	IBI	Time between two subsequent R-peaks in seconds
Frequency Ratio	RATIO	Ratio between the low frequency (LF) and the high frequency (HF) bandpower

The PSD estimation can be performed using many methods like Fast-Fourier Transform (FFT) and autoregressive modeling (AR). But, since this method requires the analyzed signal to be stationary and uniformly sampled, which is not the case with IBI signals, the Lomb-Scargle periodogram (LSP) was utilized. To compute the LSP, the MATLAB function *plomb* was used. LSP does not require resampling or stationarity signals, it only uses available data estimating the frequency spectrum and the LF/HF ratio more accurately by performing a least squares fit of sinusoids to the data, while the FFT periodograms overestimate the LF/HF ratio because of the resampling process, which adds to the LF component and reduces the HF content [78]. After obtaining the LSP, the *bandpower* function is used to return the average power contained in the HF and LF band. The average power is computed by integrating the LSP estimate, where the integral is approximated by the rectangle method. Subsequently, the RATIO is calculated by dividing the LF bandpower by the HF bandpower, where its typical value is between 1.5 and 4.5 [79].

4.1.2 Electrodermal Activity

Electrodermal activity (EDA), or galvanic skin response, is a biosignal that is associated with the sympathetic nervous system activity. This signal indicates that the skin's electrical features variated, as a result of sweat glands' activity caused by various internal or external stimuli, which is translated into conductivity [80]. This signal is obtained by measuring the voltage between two electrodes across which a current is applied, and its most common unit is the microSiemens (μS). The EDA acquisition with BITalino was made with two silver/silver chloride (Ag-AgCl) electrodes[4]. These electrodes can be placed in several ways like AB (index - middle finger), AC (index - 3rd finger) and DE (thenar eminence), all of this showed in Figure 4.12.

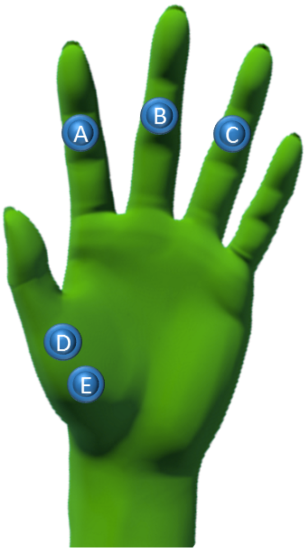


Figure 4.12: Possible EDA electrodes dispositions

[4] http://www.civicview.org/projects/

Since all of this locations bind the electrodes to the user's hand, the choice of disposition doesn't matter as all of them hamper the normal movement of the hand. So, for this dissertation, the electrodes were placed in the index and middle finger of the non-dominant hand, in order to reduce the negative impact of the worker's activity.

4.1.2.1 EDA Preprocessing

Similar to the ECG, EDA also needs to be converted to its standard units. To do that, first the Sensor resistance in megaOhms(MOhm), the R_{MOhm}, is obtained using the Equation 4.3, where ADC is the value sampled from the channel and n the number of bits of the channel (10 bits). All of these values and the converting equation can be seen in the EDA sensor Data Sheet, in appendix C. With the sensor resistance value, it is possible to obtain the EDA value in micro-Siemens(uS), as represented in Equation 4.4.

$$R_{MOhm} = 1 - \frac{ADC}{2^n} \tag{4.3}$$

$$EDA(uS) = \frac{1}{R_{MOhm}} \tag{4.4}$$

Following the EDA unit conversion, the noise removal was performed. For smoothing the signal, an exponential smoothing function with α = 0.05 was used to remove high frequency noise, as shown in Equation 4.5, where x_t is the contribution of new input data, and $(1-\alpha)F_{t-1}$ adds inertia from previous data, while α represents the smoothing factor (0<α<1).

$$F_t = \alpha x_t + (1-\alpha)F_{t-1} \tag{4.5}$$

After unit conversion, to eliminate high-frequency noise, a second-order low-pass Butterworth filter with a cutoff frequency of 1 Hz was applied to the signal. The EDA signal before and after smoothing and filtering the noise can be seen in Figures 4.13 and 4.14.

For posterior stress and fatigue detection, another essential part of the EDA processing is a reliable SCR detection. To properly detect SCRs, the method used by Kim *et al.* [80] was implemented. Initially, the original signal was downsampled to 20 samples/second in order to increase processing speed. After this, the approximate first time-derivative of the filtered EDA

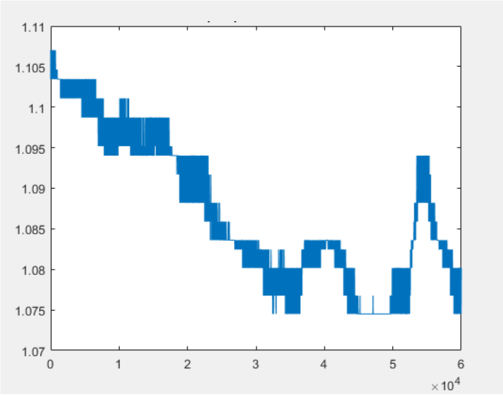


Figure 4.13: EDA signal preprocessed

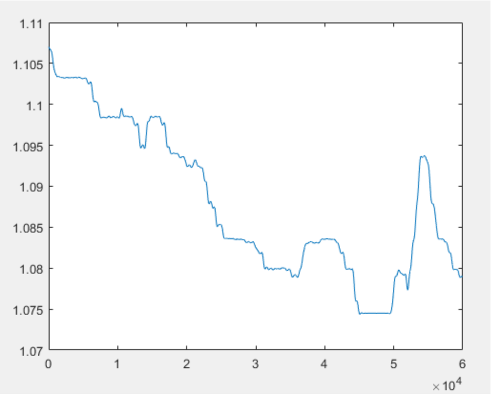


Figure 4.14: EDA signal without noise

signal was obtained using the *diff* MATLAB function, followed by a 20 point Bartlett window convolution, using the *bartlett* MATLAB function.

This procedure generated the output waveform shown in Figure 4.16 for the input signal shown in Figure 4.15. The SCR is detected when the first derivative sign changes, i.e, the detection occurs when two consecutive zero-crossings are identified (from negative to positive and positive to negative).

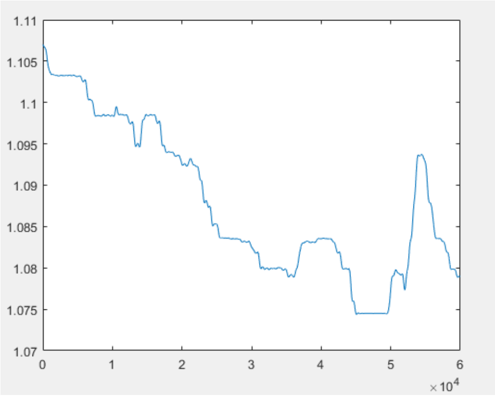


Figure 4.15: EDA signal before resample, convolution and first-derivative

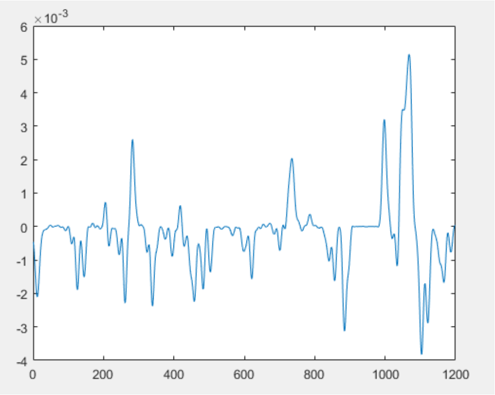


Figure 4.16: EDA signal after resample, convolution and first-derivative

4.1.2.2 EDA Features Extraction

A general problem when detecting SCRs is determining apriori if a given SCR is event-related (ER) or nonspecific (NS). It exists a risk to include NS-SCRs into the analysis for ER-SCRs if the criteria used is too sloppy, or the risk of missing many ER-SCRs if the criteria are too strict, ending up discarding important ER-SCRs. So, with this in mind, the SCRs detected with less than

15% of the maximum SCR amplitude in the segment were discarded, as implemented in by Kim *et al.* [80]. After several tests, this method showed reliable detection of ER-SCRs. Two features were acquired from the EDA signal, the SCR amplitude mean, i.e., the average of the amplitude of the detected SCRs, and the number of detected SCRs, meaning the number of the detected SCRs in the EDA signal. A review of this features can be found in Table 4.2.

Table 4.2: Features extracted from the EDA signal

Feature	Abbreviation	Meaning
SCR_number	SCR_nr	Number of relevant SCR per minute
SCR Amplitude Mean	SCR_ampl	Mean amplitude of the relevant SCR

4.1.3 Electromyography

Low amplitude bioelectrical signals sent from motor control neurons trigger muscle activation, which can be used as a reliable measure of the activation level of the muscle [81]. EMG allows these electrical signals to be translated into numerical values. With BITalino, a surface EMG (sEMG) was performed, this type of EMG uses a bipolar differential front end for enhanced signal to noise ratio (SNR). This means that, while a three lead accessory is used to obtain EMG sensor data, two leads correspond to the common positive and negative voltage and the third lead is a reference lead. So, measurements from each of the negative/positive poles are subtracted, providing a 1-D time series [82].

Most of the EMGs performed for stress and fatigue detection uses facial muscles, inferring the individual's state from his facial expressions, which in an industrial environment is unpractical. So, the muscle chosen was the trapezius, as used by Schleifer *et al.* [83] and Wijsman *et al.* [84] for stress verification. The electrode disposition was based on the recommendations found in the European R&D SENIAM (Surface Electromyography for the Non-Invasive Assessment of Muscles) project website[5], which is represented in Figure 4.17[6].

4.1.3.1 EMG Preprocessing

The first step of processing the EMG data is its normalization, where Equation 4.6 was used. It converts the values sampled from the EMG channel (ADC) to EMG value in milliVolts (EMG_{mV}),

[5] http://seniam.org/

[6] https://www.biofeedback-tech.com/articles/2016/6/21/back-pain

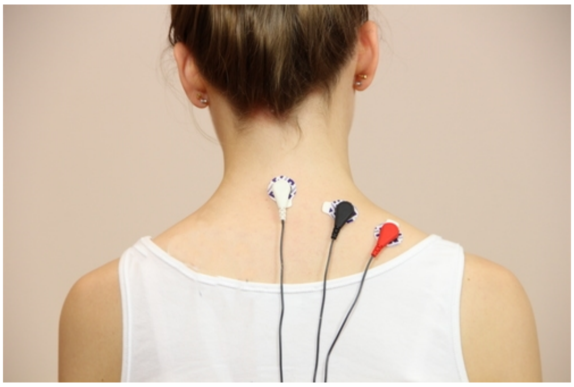


Figure 4.17: Trapezius EMG electrode placement

with n as the number of bits in the channel (10 bits), Vcc the operating voltage (3.3V) and G_{EMG} the sensor gain (1000). All these values can be seen in the EMG data sheet on appendix D.

$$EMG_{mV} = \frac{\left(\frac{ADC}{2^n} - \frac{1}{2}\right) Vcc}{G_{EMG}} \times 1000 \tag{4.6}$$

Afterward, the data is passband filtered, passing frequencies from 10 to 400 Hz (the bandwidth of the BITalino EMG sensor), using an eighth-order Butterworth filter as used by Schleifer *et al.* [83]. After that, the EMG signal was filtered using the same FIR filter as implemented for ECG to remove residual power line interference. As the EMG is performed on the left trapezius muscle, there is contamination of the ECG signal due to the proximity of the electrode placement to the heart and the volume conduction characteristics of the ECG through the torso [85]. To remove this ECG interference, it was used a highpass Butterworth filter with a cutoff frequency of 30 Hz, as it was demonstrated by Drake and Callaghan [85]. They showed that it provided the optimal balance between ease of implementation, time investment, and performance across all contractions and heart rate levels for the EMG levels. The EMG signal before and after the methods of noise removal is represented in Figure 4.18 and Figure 4.19, respectively.

4.1.3.2 EMG Features Extraction

Several features were calculated from the EMG signal, which are presented in Table 4.3. To retrieve these features, initially, the root mean square (RMS) for segments of 1 minute was calculated individually, then the static (10th percentile), median (50th percentile) and peak (90th

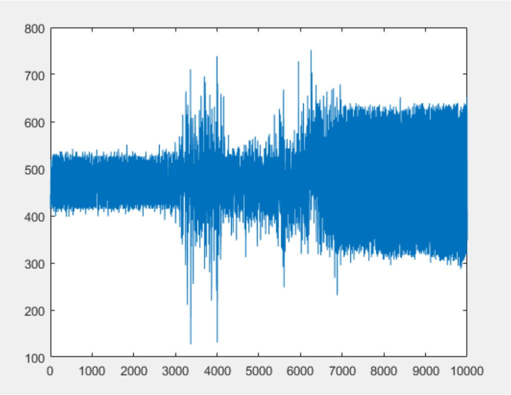


Figure 4.18: EMG signal before processing

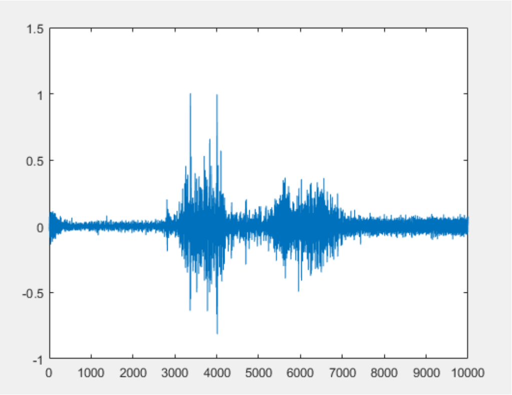


Figure 4.19: EMG signal after processing

percentile) load were also determined. To enable this, the calculated RMS values were ordered, finding the values allocated to each percentile.

The mean and the median frequency of the EMG signal were also extracted. For each segment, the Lomb-Scargle periodogram (LSP) was calculated (as used for the Frequency Ratio of the ECG in Section 4.1.1.2). After obtaining the LSP, the mean frequency was acquired, by calculating the mean frequency of the magnitude of the frequency spectrum and the median frequency. This was determined by calculating the median value, where the surface of the magnitude of the frequency spectrum on the left side equals the right side [84].

Table 4.3: Features extracted from the EMG signal

Feature	Abbreviation	Meaning
Root mean square	EMG_{RMS}	Root Mean Square (RMS) value
Static load	EMG_{Static}	10th percentile of rank ordered RMS values
Median load	EMG_{Median}	50th percentile of rank ordered RMS values
Peak load	EMG_{Peak}	90th percentile of rank ordered RMS values

Contrarily to the features extracted from EDA and ECG, the EMG features revealed themselves non-suitable for stress and fatigue inferring. This is true because, during signal acquiring, it was observed that all the features increased not only with emotional stimuli but also with any movement. A slight movement was enough to introduce noise to the signal, from simple muscular contractions due to activity execution, to minimal movements originated from body position adjustment. It was tried to surpass this setback by setting a default muscular contraction value for the individual being monitored, but the contractions, due to induced stress, most times revealed the same values. Considering that in a cooperative human-machine interaction in the industrial environment there will be constant movement from the operator while performing tasks, this feature had to be removed from the classifier, has it would increase the number of false positives in the stress and fatigue detection.

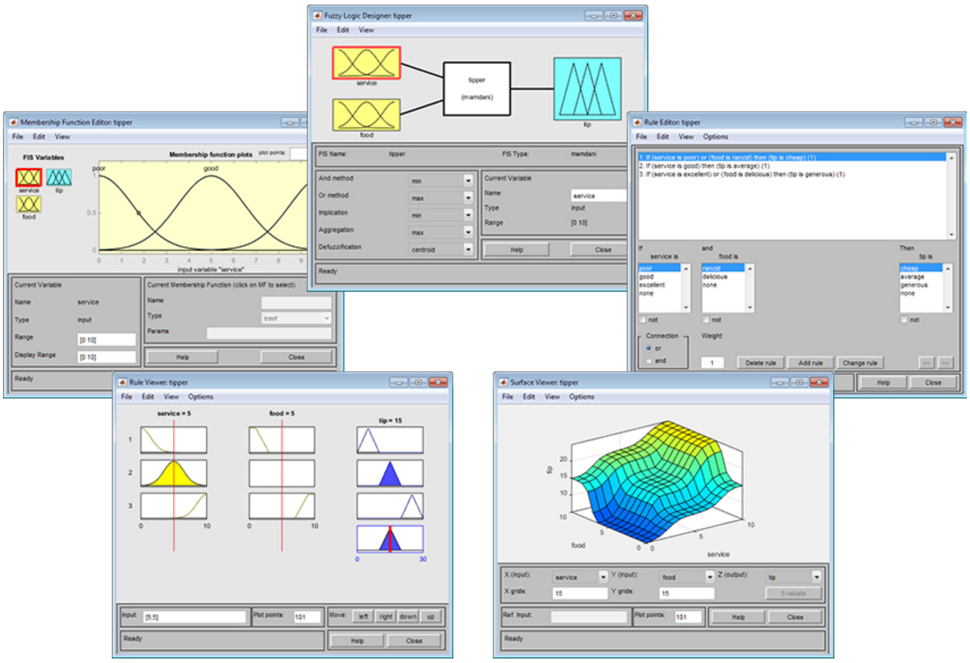
4.1.4 Fuzzy Classifier

After acquiring the biosignals and calculating the important features from those signals, it is necessary a classification technique to combine this features in a way that can allow stress and fatigue inferring. Since the transition from a low stress or low fatigue level to a high level is progressive over time, these levels cannot be treated as classical sets, since it would either include a given feature or completely exclude it. Therefore, the classification to whether a certain feature belongs to a high stress or low-stress state, or high fatigue or low fatigue state, can not be binary [32]. Since fuzzy logic is used to represent continuous processes that cannot be separated into discrete segments, with continuous variables like the features retrieved from the operator, a Fuzzy classifier was the method chosen for this implementation. Also, given the quantity of information provided by the features, the use of fuzzy logic simplifies the model by reducing the amount of data, as it quantifies the data with linguistic variables, like low, normal or high, representing an ample range of values.

In order to implement the Fuzzy method, MATLAB Fuzzy Logic Toolbox[7] was used. This toolbox allows the user to establish multiple input and output variables and membership functions of different shapes, having as main components an FIS editor, a membership function editor, a rule editor, a fuzzy inference viewer, and an output surface viewer, as represented in Figure 4.20.

The FIS editor displays the general information about the inference system, while the membership function editor allows the display and edit of membership functions associated with the FIS input and output variables. After defining the membership functions appropriately, the fuzzy rule editor and viewer is used for defining, viewing and editing the rule base, using one of three formats: full English syntax, symbolic notation, or an indexed notation. Finally, the Surface Viewer generates a 3-D surface from two input variables and one output, allowing the user to visualize the

[7] https://www.mathworks.com/products/fuzzy-logic/features.html



Membership Function Editor: Current Variable — Name: service; Type: input; Range: [0 10]; Display Range: [0 10]

FIS Name: tipper; FIS Type: mamdani

And method: min; Or method: max; Implication: min; Aggregation: max; Defuzzification: centroid

Current Variable — Name: service; Type: input; Range: [0 10]

1. If (service is poor) or (food is rancid) then (tip is cheap) (1)
2. If (service is good) then (tip is average) (1)
3. If (service is excellent) or (food is delicious) then (tip is generous) (1)

If service is: poor, good, excellent, none — and food is: rancid, delicious, none — Then tip is: cheap, average, generous, none

Connection: ● or ○ and; Weight: 1

Rule Viewer — Input: [5;5]; Plot points: 101

Surface Viewer — X (input): service; Y (input): food; Z (output): tip; X grids: 15; Y grids: 15; Plot points: 101

Figure 4.20: The Membership Function Editor (top left), FIS Editor (center), Rule Editor (top right), Rule Viewer (bottom left), and Surface Viewer (bottom right)

rule base. The design and implementation of the fuzzy model in MATLAB used to obtain stress and fatigue levels involved performing the two main steps: 1) Specifying the input and output variables membership functions; and 2) Defining the rule-base. Besides these two steps, MATLAB's Fuzzy Logic Toolbox executes by itself other necessary steps for fuzzy classification.

4.1.4.1 Input and Output Variables Membership Functions

A membership function defines how an element is mapped to a membership value, translating the input variable into a range of values between 0 and 1. This range of values signifies the degree of confidence to which an input belongs to a specific set (0 means the element does not belong to the fuzzy set and 1 indicates the element completely belongs to the fuzzy set). There are various types of membership functions, as they can be built from simple functions like straight lines, triangular or trapezoidal memberships, or more complex like Gaussian distribution functions and polynomial based curves. The Gaussian membership function can be specified with two parameters: mean value and standard deviation, so due to its smoothness and succinct specification was the one used for the input and output memberships.

For each input (HR, IBI, RATIO, SCRnumber, and SCRamp), the membership functions were created using the standard deviation and mean value of those inputs in three different situations: high stress, high fatigue, and total relaxation. The high-stress condition was induced by making the subject play a timed smartphone game of increasing levels of difficulty against another online player, inciting competitiveness and pressure of performance, and therefore stress. The high fatigue condition was encouraged by doing approximately 30 minutes of heavy physical effort, after which occurred a rest period of 10 minutes then measuring the inputs. The total relaxation scenario involved the individual in a stationary and comfortable form while listening to relaxing music. The relax condition was tested before all the others to ensure that they did not tamper with the neutral condition, due to possible remains of stress and fatigue levels if performed later, while the high fatigue condition was simulated after all the others, increasing the fatigue level. Each condition test was executed 20 times, for periods of 10 minutes, having a total of 20 samples for each situation, with 10 minutes of continuously acquired and processed data.

In order to normalize the data, the minimum, maximum, mean value and standard deviation (STDV) from each set of samples of each input were calculated. This was necessary in order to use the Z-score method. The Z-score method preserves range (maximum and minimum) while introducing the dispersion of the series (standard deviation) and has a range from -3 standard deviations up to +3 standard deviations. The basic Z-score formula for a sample is represented in Equation 4.7, where x_i is the sample data, $\overline{x}$ is the sample mean, s the standard deviation, and z_i is the i^{th} normalized data.

$$z_i = \frac{x_i - \overline{x}}{s} \tag{4.7}$$

After obtaining the normalized inputs through the Z-score method, the Equation 4.8 was used to set the values between 0 and 1, where x_i=(x_1,...,x_n) is the z-score normalized data and z_i is the i^{th} data obtained in values from 0 to 1.

$$Z_i = \frac{x_i - min(x)}{max(x) - min(x)} \tag{4.8}$$

The normalized mean values and standard deviations obtained for each input in each simulated situation are presented in Table 4.4. These values were used to define the Gaussian membership functions that suit the distribution of the inputs.

Table 4.4: Mean Value and Standard deviation of the normalized inputs for each condition simulated

	High Stress		**High Fatigue**		**Relaxation**	
	Mean	STDV	Mean	STDV	Mean	STDV
HR	0,712	0,087	0,112	0,069	0,391	0,043
IBI	0,095	0,041	0,731	0,135	0,273	0,042
RATIO	0,841	0,101	0,151	0,112	0,538	0,083
SCRnumber	0,470	0,185	0,052	0,071	0,060	0,058
SCRamp	0,206	0,226	0,078	0,185	0,110	0,193

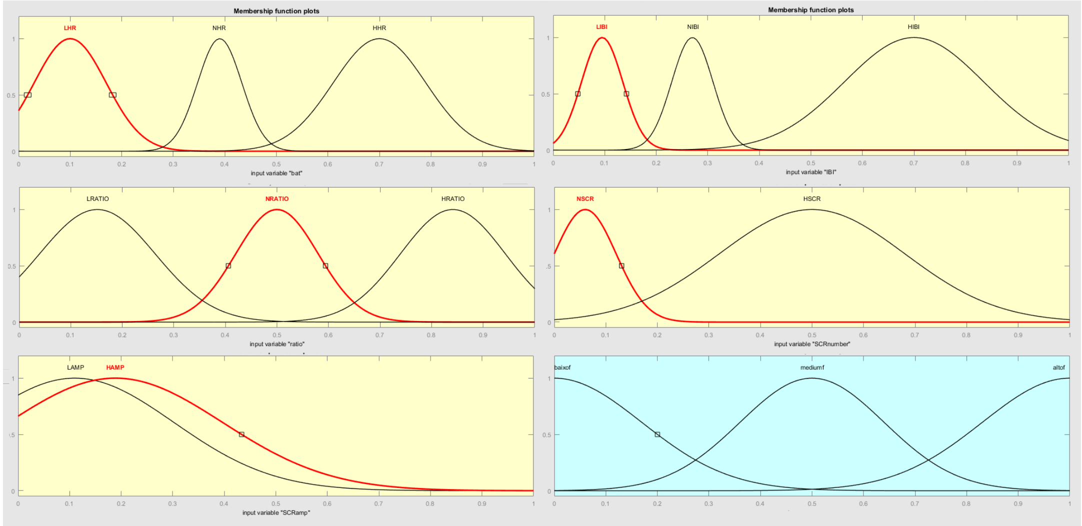


Figure 4.21: HR (top left), IBI (top right), RATIO (center left), SCRnumber (center right), SCRamp (bottom left) and stress or fatigue memberships (bottom right)

The Gaussian membership functions are represented in Figure 4.21, where the ones with an orange background are inputs and the one with the blue background is membership used for both outputs. The membership functions for the two outputs (stress and fatigue) were distributed evenly through the Gaussian function, being defined with three memberships: low, medium and high.

4.1.4.2 Defining the Rule-base

Once the input and output variables and memberships are defined, the rule-base needs to be created, being composed of IF-THEN rules that connect the input and output fuzzy sets. These rules are constituted by several inputs connected by the logic “AND” or “OR” that are applied to establish the FIS with fuzzy operations. To create the rules for this fuzzy a study of all possible combination of inputs was performed, obtaining a total of 27 rules. These rules were grounded in the theory of how the physiological metrics retrieved from the signals relate to the psychological concepts of stress and fatigue, described in Section 3.1. All of the rules created are annexed in Appendix E. The rule base is an essential component, as the stress and fatigue levels will be obtained rigorously in accordance with the pattern of the rule base, and therefore defective patterns will result in erroneous levels.

4.1.4.3 MATLAB's Fuzzy Logic Toolbox Inner Steps

Besides these steps performed, MATLAB's Fuzzy Logic Toolbox executes by itself other necessary steps for fuzzy classification:

- Fuzzification of the inputs - The inputs are fuzzified from the crisp values of the variables to membership values before evaluating the defined rules. Independently of what the input variable describes, the output is the degree of membership. So, having as inputs the crisp normalized values of IBI, HR, RATIO, SCRnumber, and SCRamp, and as output a fuzzy degree of membership in the corresponding set (stress or fatigue), a fuzzification is performed.

- Fuzzy Inference - The fuzzy inference is a process of mapping an input to an output space based on the test of all IF-THEN rules, membership functions, and fuzzy logical operations. All the rules are combined and evaluated and the output of each rule is aggregated into a single fuzzy set whose membership function assigns a weight for every output. Generally, three types of fuzzy inference methods are used: Mamdani fuzzy inference, Sugeno fuzzy inference, and Tsukamoto fuzzy inference. For this Fuzzy classifier, the Mamdani-Type Fuzzy inference process was the one used as it is more intuitive, has widespread acceptance, and it is well suited to human input. In Mamdani inference the IF-THEN rules obtain an output value defined by a fuzzy set, that will be reshaped by a matching number, is required to execute a defuzzification after combining all of these reshaped fuzzy sets.

- Defuzzification of the outputs - This process transforms its input (the resulting aggregated output fuzzy set) into a single number, and can employ methods like centroid, bisector, middle of maximum (the average of the maximum value of the output set), largest of maximum and smallest of maximum. The defuzzification method used was the centroid calculation, which returns the center of the area under the curve.

In resume, the MATLAB Fuzzy Logic Toolbox allows creating a fuzzy classifier that only requires a previous definition of the input and output memberships, as well as the rule-base. Then, using the *readfis* function, the defined fuzzy inference system is read from a file. After this, it follows the *evalfis* function that performs the fuzzy inference calculations, returning the stress and fatigue levels in an output matrix of size 1x2, based on a 1-by-5 input matrix (HR, IBI, RATIO, SCRnumber, and SCRamp) and the FIS file. The simplified version of the Fuzzy classifier implemented in MATLAB is demonstrated in Figure 4.22.

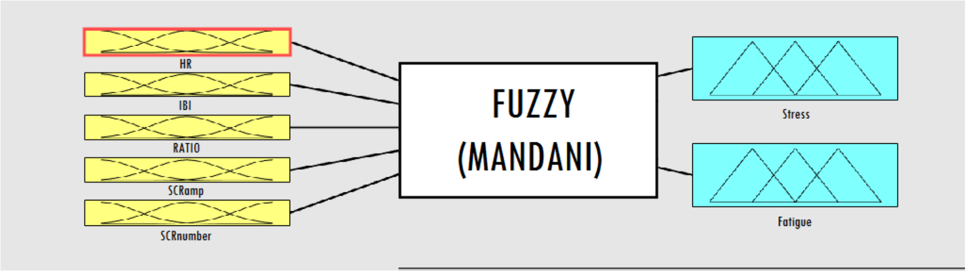


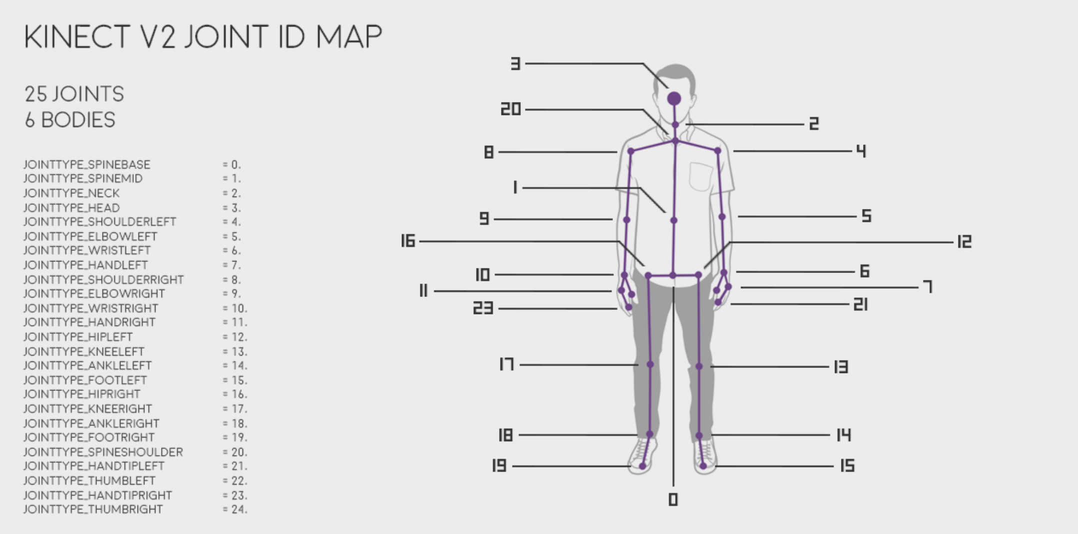
Figure 4.22: Inferring stress and fatigue from physiological data. The system used 12 rules to transform the 5 inputs into the 2 outputs

4.2 Position Monitoring & Motion Tracking

For the adaptation of the robotic manipulator to the operator's task order preference, human position monitoring is essential. The idea is to track how the different industrial operators performed their actions, allowing the machine to perceive automatically where it should execute its task in the cooperative process, according to the operator. In order to perceive the operator's motion and body position the Microsoft's Kinect v2 sensor (see Section 3.2.4) and Software Development Kit (SDK), 2.0 were used.

4.2.1 Body Tracking

The Kinect SDK 2.0 provides several code samples, enabling the creation of applications using the Kinect sensor, connected to machines running Windows operating system. One of the most pertinent parts of the Kinect SDK for this dissertation is its support for tracking the human body, which allows to track up to 6 people, and 25 joints per body. The joints supported by the Kinect v2[8] are shown in Figure 4.23.



KINECT V2 JOINT ID MAP

25 JOINTS
6 BODIES

JOINTTYPE_SPINEBASE	= 0.
JOINTTYPE_SPINEMID	= 1.
JOINTTYPE_NECK	= 2.
JOINTTYPE_HEAD	= 3.
JOINTTYPE_SHOULDERLEFT	= 4.
JOINTTYPE_ELBOWLEFT	= 5.
JOINTTYPE_WRISTLEFT	= 6.
JOINTTYPE_HANDLEFT	= 7.
JOINTTYPE_SHOULDERRIGHT	= 8.
JOINTTYPE_ELBOWRIGHT	= 9.
JOINTTYPE_WRISTRIGHT	= 10.
JOINTTYPE_HANDRIGHT	= 11.
JOINTTYPE_HIPLEFT	= 12.
JOINTTYPE_KNEELEFT	= 13.
JOINTTYPE_ANKLELEFT	= 14.
JOINTTYPE_FOOTLEFT	= 15.
JOINTTYPE_HIPRIGHT	= 16.
JOINTTYPE_KNEERIGHT	= 17.
JOINTTYPE_ANKLERIGHT	= 18.
JOINTTYPE_FOOTRIGHT	= 19.
JOINTTYPE_SPINESHOULDER	= 20.
JOINTTYPE_HANDTIPLEFT	= 21.
JOINTTYPE_THUMBLEFT	= 22.
JOINTTYPE_HANDTIPRIGHT	= 23.
JOINTTYPE_THUMBRIGHT	= 24.

Figure 4.23: Kinect v2 Joint Map

The body tracking function was the one used to monitor the operator's task order preference, tracking, in this use case, his right hand during the manufacturing process, in order to detect in which region in the shared area of production he is acting. The body-tracking feature is built in the depth data processing, matching the raw depth data from the sensor with the sampled trained data (data with different heights, clothes, and other factors labeled with each body part) to identify which part of the body it belongs to [86].

This process includes several steps. Initially, the sensor identifies the human body object by matching each pixel of depth data with the data used for training; secondly the body parts are labeled by creating segments that are developed by matching similar probable data, as shown in Figure 4.24 b), and finally, a trained decision tree, specifically a Decision Forrest, is used to match the data to a type of body. The Decision Forrest used is composed of nodes that are data labeled with body parts, with several different characteristics, where every pixel of depth data passes through to match with body parts. When some data is matched, the sensor starts creating body segments and, when identified, the 3D joint points with the most probable matched data are positioned. In the Figure 4.24[9] is shown the positioning process of the joints, as described

[8] https://vvvv.org/documentation/kinect

[9] based on http://www.i-programmer.info/news/105-artificial-intelligence/2176-kinects-ai-breakthrough-explained.html

above [87].

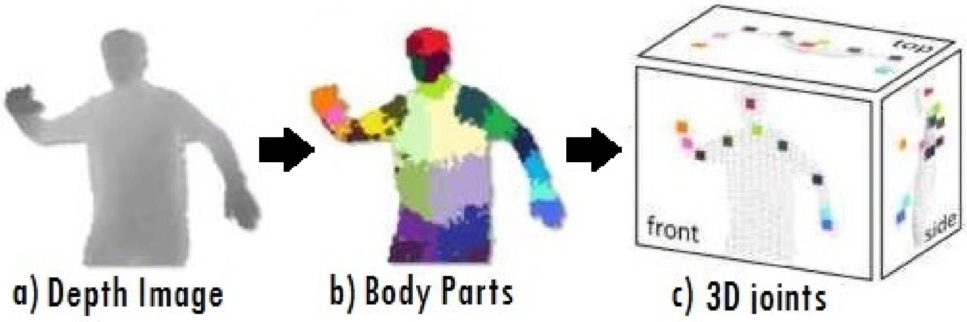


Figure 4.24: Joint positioning process

The joint positions that were defined are measured by X, Y, and Z coordinates as shown in Figure 4.25. While X and Y define the position of the joint, Z represents the distance from the sensor, all measurement in meters. This system of axis is projected on the subject, consequently having X and Y as negative or positive values, and the Z coordinate always as positive. In order to obtain the proper coordinates, Kinect calculates the front, left and top view from the same image, defining the 3D body [88]. The three views can be seen in Figure 4.24 c).

With Kinect SDK 2.0 is possible to easily access the body joints since each one of the 25 possible joint positions are identified by a name, as shown in Figure 4.23. To access the operator's right hand and understand what task he is performing, a C# code was created that uses the monitoring of X, Y and Z coordinates of the right hand to return the area in which the operator is performing its task in real time.

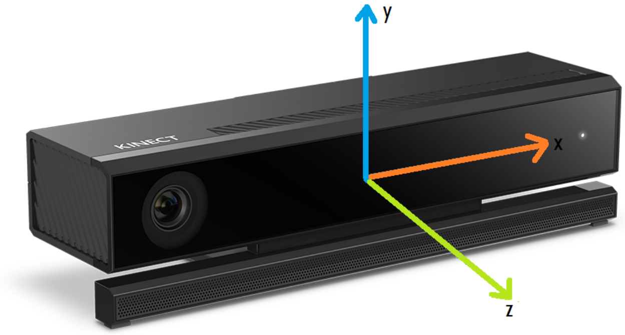


Figure 4.25: Kinect coordinate system

In the code, after initializing the Kinect sensor, the *MultiSourceFrameReader* class is used to access the body and color streams, followed by the creation of a frame reader method, which is called whenever a new frame is available. By using the *BodyFrameReader*, is possible to access the *BodyFrame* that contains all the computed real-time tracking information about the people that are detected by the sensor, using the method *GetAndRefreshBodyData*. It's important to check for null values in the frames acquired, as Kinect provides 30 frames per second, and anything could be null or missing, leading to program crashes.

After having access to the body that Kinect identifies, its state must be checked to know whether it is tracked or not before acquiring the right-hand coordinates. The *HandRight Jointtype* is used to detect the right hand, as shown in Figure 4.23, retrieving its X, Y and Z coordinates from the joint position in the camera space. With the coordinates acquired in real time, the next step consists of checking if their position in the axis match the range of values defined to limit the areas associated with certain tasks in the production process, returning the corresponding area in where the hand is located. The area returned is acquired each 500ms, having a sampling rate of 2Hz, and it can result in different integer values:

- area is 0 if the operator is not performing a task in a defined work area.
- area is a positive integer number if the operator is performing a task in a defined work area.

Like any other sensor, the Kinect also has physical limits, namely regions outside of the camera scope. There are preferable spots where an optimal experience can be obtained and consequently achieving better results when tracking the operator's hand. The fixed measures to these physical limitations and preferable spots are represented in Figure 4.26.

To make the area monitoring more reliable and stable, many factors had to be taken into account. For instance, external lighting sources affect image quality and depth sensing, given that with large amounts of natural light, body tracking becomes less reliable. This occurs due to the low contrast of the IR image and therefore depth values of 0 (unknown). The environment lighting parameter should be taken into account when performing the operator's monitoring, namely maintaining the ambient light as a controlled parameter. Also, other features such as individual's clothes or objects that the human holds during the production process can interfere with the measuring process. For example, black clothing, as well as reflective items, can meddle with the IR camera, making again the body tracking method less reliable. Likewise, to guarantee a robust tracking method, the individual should keep his arms and hands to the side of the body, and his

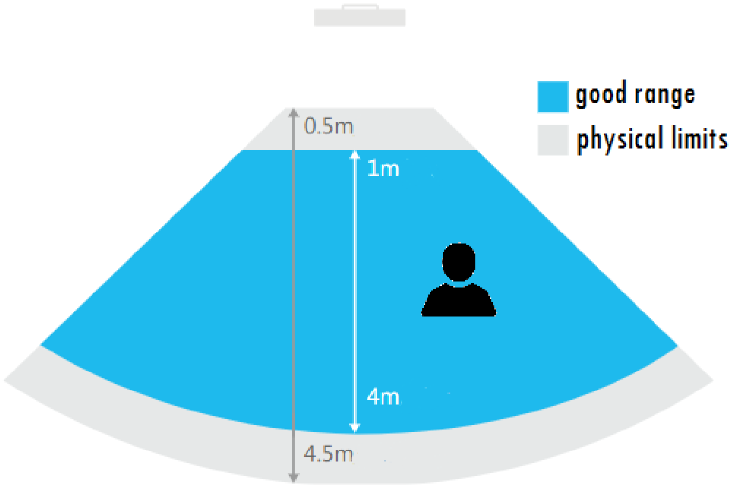


Figure 4.26: Physical limits and preferable spots for Kinect v2(Based on [6])

face to the sensor when performing actions, since hand movements in front of the body and facing sideways can be unreliable [6].

Livingston *et al.* [89] made a study of performance measurements for the Kinect Skeleton, in which tracker noise was considered one of the biggest limitations. To quantify that noise, the user was physically restrained with hidden structures in order to stand still, taking 1000 samples for the tracked body, and obtaining more noise in the right-hand joint (22.6mm) and right wrist (31.0mm) than in any other joint. Latency was also considered a problematic performance characteristic. At the normal frame rate of 30 Hz, while tracking the body, they obtained a mean latency of 146ms (maximum 243ms), while bringing the right hand down to the left, in which a mouse was held and thus the button hit. All of this limitations had to be taken into account when monitoring the operator's hand, demanding a very controlled environment as well as steady movements when performing actions.

4.3 Cooperative Human-Machine Interaction Architecture

The CHMI solution consists in the implementation of a system that represents the cooperative environment between the human operator and industrial robot, in form of a CPPS. Since the CPPS presents self-adaptation capabilities, a MAPE-K methodology was used. The overall implementation is resumed in Figure 4.27.

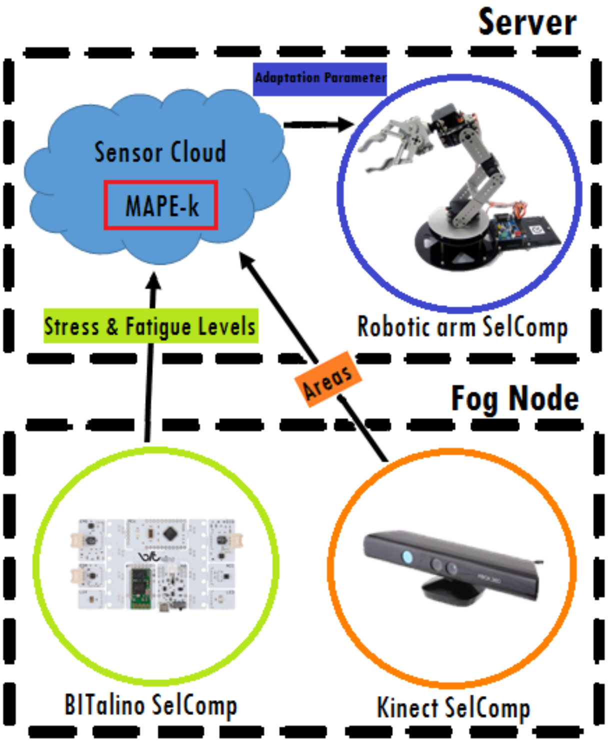


Figure 4.27: CHMI implementation scheme

So, in a nutshell, this solution is mainly composed by a CPPS with an adaptive module (MAPE-k), a Cloud infrastructure (SELSUS project Sensor Cloud), Fog Nodes, and three different SelComps: BITalino SelComp, Kinect SelComp, and Robotic Arm SelComp, where the first two provide data from the monitored operator and the last one serves as actuator. The BITalino Selcomp and the Kinect Selcomp are implemented inside the Fog node where all the data is processed and information inferred, being posteriorly sent to the Sensor Cloud via HTTP. The adaptive module (MAPE-k) is called every time the Cloud receives values from the SelComps, having as inputs the data from the SelComps and as outputs the adaptation parameters determined in the case of necessary adjustments in the process, these outputs are then sent to the Robotic arm SelComp to actuate.

All these concepts and its methods of implementation are described in the next subsections.

4.3.1 Physical architecture (CPPS)

The CPPS architecture used is based on the one described for production systems in the Chapter 2.2.

- The Smart connection level is where all physical (sensing and actuating) devices from the process are located, namely the Kinect, BITalino and Robotic arm. This level is characterized by the collection of all the data from the sensors, such as biometric signals and body motion.
- The Data-to-information conversion level consists in inferring the information needed for the system adaptation, such as the stress and fatigue levels and the area from, respectively, the biometric data and the process from the operator's position. This information results from a fuzzy classifier, which correlating the metrics retrieved from the BITalino's ECG and EDA sensors, and an algorithm that corresponds the depth measures from the Kinect to certain areas were used.
- For the Cyber level, an agent-like system based on Smart Components is utilized, virtualizing all the physical devices from the previous level in a Cloud infrastructure. Both Smart Component and Cloud infrastructure result from the R&D European project SelSus - Self Sustaining Manufacturing Systems, where Smart Components are materialized into SelComps and the Cloud infrastructure developed is called Sensor Cloud.
- In the Cognition Level, the same Sensor Cloud is used, as it allows to graphically represent all the historical and real-time data from the monitored operator, transferring this knowledge to the users, using a web-based interface.
- In the Configuration level, the knowledge of the monitored process in terms of the operator's psychological state and task preference is used by the adaptive module (MAPE-k) to decide which values should be sent in the adaptation parameters. These parameters are then sent to the robotic arm as complete actions for direct actuation in the environment.

The proposed CPPS is composed by three physical devices: 1) the BITalino (detailed in Section 3.2.3); 2) the Kinect (detailed in Section 3.2.4); and 3) the Robotic arm. The robotic arm used in this scenario is the Linxmotion's AL5d and possesses four axis + gripper, with a range of motion of 180 degrees per axis, lifting up to 368.5 grams. It has a local closed loop as servo motion control and a SSC32U servo controller. All these physical devices (Kinect, BITalino, and Robotic arm) are virtualized into Smart Components or, in the scope of the SelSus project, SelComps. SelComps consist in the encapsulation of physical devices, characterized by their modules for sensor integration and posterior sensor data local processing. This data is later sent to the Sensor Cloud, for high-level data analytics. This Sensor Cloud provides user interfaces that allow the users to monitor and control the system's sensors via Web browsers [90].

In order to create the SelComp, a XML file format, entitled Selcomp Self Descriptor (SSD), is sent via HTTP to the Sensor Cloud, before starting the data acquisition/actuation. This SSD file represents the identification and capability characterization of each SelComp that is integrated into the CPPS. This file includes the SelComp name (that will be displayed in the Cloud), type, ID (unique or each device's), IP and port, allowing to associate the data with the physical device. After sending the SSDs, all SelComps are created in the Sensor Cloud. The BITalino Selcomp acquires data from the ECG and EDA sensors, that after processed are used to infer stress and fatigue levels in real time, while the Kinect Selcomp obtains the operator's work area from the position of his right hand. All SelComps are implemented inside a Fog Node, which extends the Sensor Cloud data analytics capabilities to the devices level. Devices are connected to these Fog Nodes into several forms, namely via USB in the case of the Kinect and via Bluetooth for BITalino. Ultimately, the SelComps in the Fog Nodes communicates the sensed data remotely with the Sensor Cloud.

In order transmit each measurement of stress and fatigue, as well as the areas, the SelComps send another XML-type file every time a new value is acquired, namely the Payload. The payload contains the data acquired from the physical devices, in a simple and XML schema, which is transmitted to the Sensor Cloud with the corresponding timestamp. In this file, besides the value and timestamp, the service type is also defined, which can be "DIAGNOSIS", "MAINTENANCE" and "OBSERVATION" state. With all the SelComps implemented in the Sensor Cloud and the SSD and Payload files defined, the Physical component of the CHMI architecture is complete and therefore the implementation of the functional architecture (MAPE-k) follows.

4.3.2 Functional Architecture (MAPE-k)

As described in Section 2.3, a common reference model used in self-adaptive systems is MAPE-K. So, for the functional part of the CHMI architecture, a MAPE-k feedback loop was applied, which allowed the adaptive actuation of the robotic manipulator, performing different actions with distinct parameters according to the human in the cooperative environment. Figure 4.28 shows the described MAPE-K implementation.

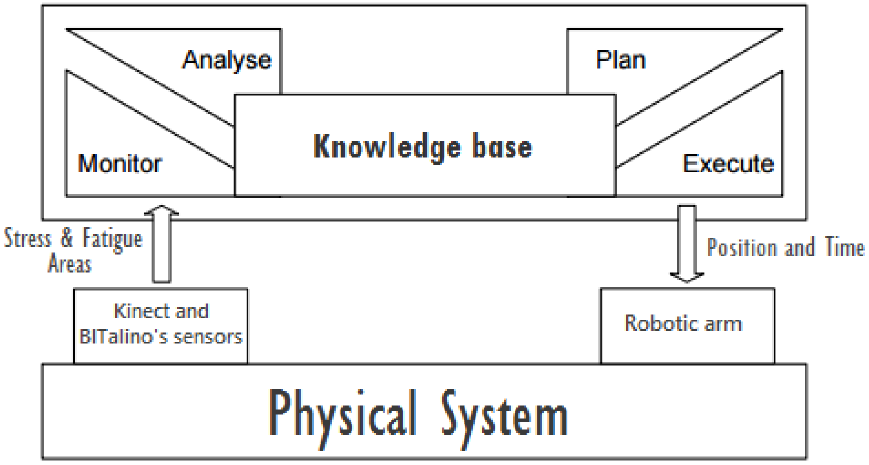


Figure 4.28: MAPE-K implementation

While monitoring the operator, MAPE-k uses the stress and fatigue levels as well as the areas obtained, to plan the necessary adaptive measures. These measures consist on sending commands that lead to executing a specific action through the robotic arm, in the case of existing a condition that drives to the need of adaptation, like high-stress levels or high fatigue or task area. The adaptation parameters defined are the robotic arm's execution time while performing its tasks, and the position in which that task should be performed. The task execution time is adapted according to the operator's psychological levels, while the identification of the area depends on the values detected by the Kinect sensor, that is the position where the human operator is performing his task. The MAPE-k function is implemented as a module in the Sensor Cloud, being called every time a new value of stress and fatigue or operator's area is received in the Cloud, returning the adaptation parameters task area and time, which are sent to the robot.

When data is received, the code analyzes it, using a set of conditions to conclude if a command to execute an action needs to be sent to the robotic arm. If the MAPE-k decides that no action is necessary, the code returns -1 as the task area and time values. Since the psychological levels returned from BITalino were acquired at a frequency of 1 Hz and the areas from Kinect at 2 Hz, the synchronization when receiving values was not possible, so the use of the Sensor Cloud database was essential in order to store the stress and fatigue levels of the user, when an area associated with a task was not acquired.

So, in cases where only the values of stress and fatigue are received, the outputs of the MAPE-k are sent as -1, followed by the call of a function that stores the stress and fatigue values in the database. This storing and posterior acquiring of levels in the database did not compromise the adaptive nature of the process since stress and fatigue suffer very slow and gradual variations. A read database function is called every time the MAPE-k function runs. Conversely, when only the area is received, the MAPE-k checks if the user is in that position for the time taken for the user to complete its task. If the area was received for less than that time, the function returns -1 and no task is executed, but if the time condition checks, the most recent value of stress and fatigue from the database is used, determining in which task area the robotic arm should perform its action and with what execution time. These values are then returned as JSON, to posteriorly be sent to the actuator.

The execution time is calculated according to stress and fatigue levels, by multiplying the maximum time the robotic arm can achieve (1600ms) by a "discomfort factor", *k*, that takes its highest value (1) for cases of High/High stress and fatigue classifications combined, and minimum value (0.375) for Low/Low. This discomfort factor decreases with the reduction of stress and fatigue levels, allowing the process to adapt the time of task execution of the robotic manipulator. This enables an improvement in the operator's work conditions, giving him more time to recover between tasks. In cases where the classifications of stress and fatigue switched, for instance, Low stress/High fatigue and High stress/Low fatigue, or Medium Stress/Low fatigue and Low stress/Medium fatigue and so on, the *k* value applied was the same, given that when in the same classification level, the impact of stress or fatigue in the operator's performance is practically equal (both diminish productivity and worsen the individual's work conditions). So a total of 6 different *k* values were defined, one for each Stress/Fatigue level combination.

Chapter 5

Validation & Testing

In this chapter, the global use case scenario is explained, as well as all the tests performed to validate and evaluate the proposed implementation, explaining and discussing its results.

5.1 Use Case Scenario

For simulating the cooperative interaction between the human operator and industrial equipment, the robotic arm described in Section 4.3 was used, collaborating with an individual to execute a shared production process. The process is defined by a total of 21 sequential tasks, 9 executed by the robotic arm and 12 by the subject. These tasks can be classified into two groups, according to the entity that executes them, as described in Table 5.1.

Table 5.1: Possible tasks in the process for each entity

Robotic Arm	Human operator
Place screw in Box 1 (3x)	Place Box 1 (1x) / Rotate screw in Box 1 (3x)
Place screw in Box 2 (3x)	Place Box 2 (1x) / Rotate screw in Box 2 (3x)
Place screw in Box 3 (3x)	Place Box 3 (1x) / Rotate screw in Box 3 (3x)

Regarding the use case scenario, a human operator is sited in front of the work area, executing the tasks of placing a box in the corresponding area. In the end of the process, the operator should place all the boxes and can freely choose how to do it, e.g., placing in sequence boxes 1-2-3, or any other order. While executing the tasks, the human operator's ECG and EDA are being monitored, as described in Subsections 4.1.2 and 4.1.1. After a human task regarding placing a box, the robotic arm immediately executes one of its tasks, which consists of putting a screw inside a box. In order to put the screw in the right box, the robotic arm should be aware of what was the area that the human chose to place a box in. Right after the robotic arm task execution, the human executes again one of this tasks, which can be placing another available box in the corresponding area or rotating the screw placed by the robotic arm. In total, there are three screws to place and rotate in each box.

After the individual's choice of task in a certain box, the robot will acknowledge that choice and perform the placement of the next screw in the right box. All of the robotic arm's tasks are executed within the same normal execution time. However, in order to test the adaptation to the stress and fatigue levels of the operator, this execution time is modified in accordance with those levels, as mentioned in Subsection 4.3.2. The use case scenario and its components can be seen in Figure 5.1.

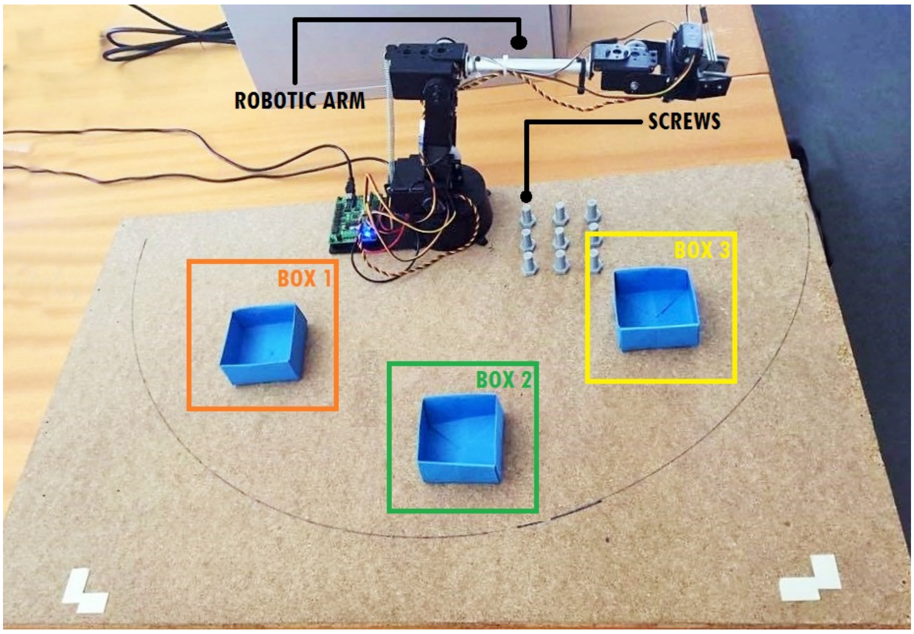


Figure 5.1: Use case scenario and its components

5.2 Experiments and Results

5.2.1 Experiments

Since the robotic arm reacts directly to the values returned from the operator's monitoring, the accuracy of the proposed solution is straightforwardly related to the accuracy of the stress and fatigue classifier as well of the position monitoring and motion tracking. So to assess the solution's reliability the following experiments were performed:

1. Stress and Fatigue Classifier evaluation:

   Two subjects (A and B) were tested in two different conditions - induced stress and induced fatigue - where each subject performed 15 tests of 5 minutes for each psychological level in each condition. The induced conditions were provoked on the subjects using the same procedures explained in Subsection 4.1.4 for high stress and high fatigue situations when acquiring the initial data.

   Considering that the stress and fatigue classifier must reach a compromise between detecting properly if the subject is under stress/fatigue and detecting when it is not, a Receiver Operating Characteristic (ROC) curve analysis was performed by evaluating the Area Under the ROC curve (AUROC), along with a accuracy, sensibility and specificity calculation. For this analysis, the True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN) were obtained by comparing the subject's self-assessment regarding his own levels of stress and fatigue with the result returned by the classifier. After each test, the subject would determine if he was under stress or fatigued if the Fuzzy classifier obtained a level greater than 0.3 (cut-off value) the detection of stress or fatigue was considered positive.

2. Position Monitoring and Motion Tracking evaluation:

   One subject was tested executing the tasks defined in Section 5.1. The individual carried out 15 tests with the duration of 2 minutes each, remaining sited within the Kinect's fixed measures for better tracking, and executing the tasks with his right hand. During the 2 minutes tests, it was attempted to equally separate the time spent in each area, alternating between the four existing areas (area 1 corresponding to box 1, area 2 to box 2, area 3 to box 3 and area 0 when out of those defined coordinates, see Figure 5.1 for the boxes' locations). It's important to refer that the user needed to always initially "calibrate" the Kinect, by opening its arms in front of the camera while sitting.

   For this evaluation four parameters were defined: True Area Detection (TAD) - when the position tracking system properly detects an area, False Area Detection (FAD) - when the system incorrectly detects an area, TAD rate ($TADR$) and FAD rate ($FADR$). Notice that the higher the $TADR$ and the lower the $FADR$, the more accurate the performance of the system.

3. Use Case Scenario testing:

   In order to conclude in what way the collaboration between the operator and the robotic arm could influence the production system, the use case scenario explained above in this section was enacted, including the monitoring systems with the adaptation module and overall CPPS. The test was performed by two subjects of different genders and lifestyles, inducing different stress and fatigue levels with the objective of testing all possible adaptation combinations (Low/Low, High/High, Medium/Medium, Low/Medium, Medium/High and Low/High). Each subject executed a total of 18 tests, three times for each type of test, i.e for each combination of stress and fatigue levels mentioned above. These tests only serve as demonstrators to the adaption according to the monitoring performed in each individual, since the results obtained will depend on the accuracy of the previously defined classifier and monitoring evaluation.

   The mean values of the times returned from the MAPE-k were calculated for every task the robotic arm performed in each box and from these values, the following evaluation parameters were obtained for each subject: Mean Execution Time (MET) in milliseconds for box 1, 2, 3 and total process for all adaptation combinations, and MET Variation ($VMET$) in relation to the no stress and no fatigue state (Low/Low combination) MET.

5.2.2 Results

5.2.2.1 Stress and Fatigue Classifier

For the Stress and Fatigue Classifier evaluation, the inferred levels were extracted during the tests from the MATLAB BITalino API to a text file. Also, in order to evaluate the data generated from the tests, the historical data stored in the Sensor Cloud was used. For the ROC curve analysis, the true positive rate (TPR) and the false positive rate (FPR) were calculated by Equation 5.1 and Equation 5.2, allowing then to obtain the ROC curve by plotting TPR against FPR.

$$TPR = \frac{TP}{(TP+FN)} \tag{5.1}$$

$$FPR = \frac{FP}{(TN+FP)} \tag{5.2}$$

Table 5.2 shows the TP, TN, FP, FN, TPR and FPR values for each subject, while in Table 5.3 those same values are calculated for each experiment combining stress and fatigue of both subjects.

Table 5.2: ROC analysis for Subject's A and B fatigue and stress detection results in each test

	Stress						**Fatigue**					
	TP	TN	FP	FN	TPR	TNR	TP	TN	FP	FN	TPR	TNR
Induced Stress Subject A	9	3	2	1	90%	60%	2	11	2	0	100%	85%
Induced Stress Subject B	12	1	2	0	100%	34%	1	10	3	1	50%	77%
Induced Fatigue Subject A	8	2	0	5	62%	100%	3	9	3	0	100%	75%
Induced Fatigue Subject B	10	1	0	4	71%	100%	1	9	4	1	50%	69%

Table 5.3: Global ROC analysis of the stress and fatigue detection for each test scenario

	Stress						**Fatigue**					
	TP	TN	FP	FN	TPR	TNR	TP	TN	FP	FN	TPR	TNR
Induced Stress Test	20	4	4	2	91%	50%	3	21	5	1	75%	81%
Induced Fatigue Test	4	18	7	1	80%	72%	18	0	9	3	67%	100%

The AUROC, calculated using the Trapezoid Rule, was then evaluated as it serves as a great indicator of the accuracy of the fuzzy classifier, being equivalent to the "probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance" [91]. Random guessing produces the diagonal line between (0, 0) and (100, 100), being equivalent to an area of 5000, so no realistic classifier should have an area lower than that value. Figures 5.2 and 5.3 represent the ROC curves for each test condition and for each classification, while in Table 5.4 the AUROC values for stress and fatigue classification in each condition are represented.

Finally the values of accuracy (ability to differentiate stress/fatigue from non-stress/non-fatigue cases correctly), sensitivity (ability to determine the stress/fatigue cases correctly) and specificity

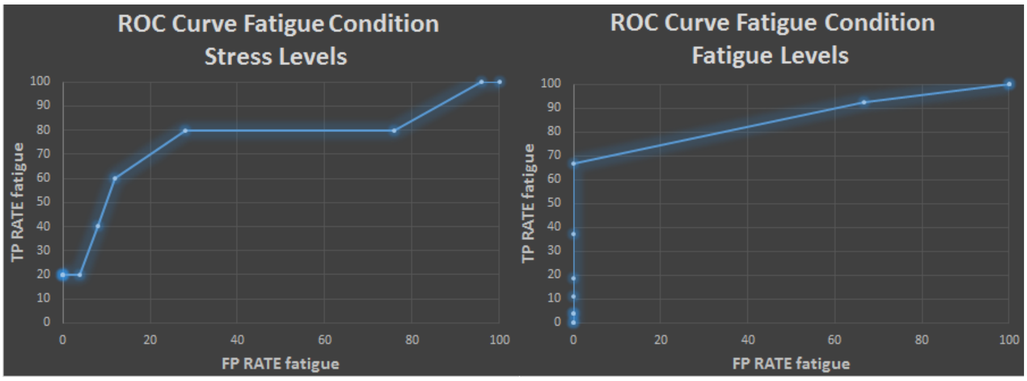


Figure 5.2: ROC curves for the fatigue condition experiment (left curve - Stress Levels inferring and right curve - Fatigue Levels inferring)

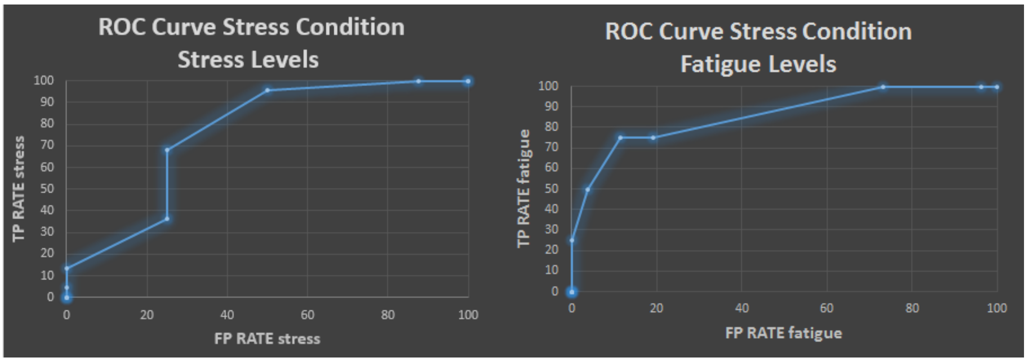


Figure 5.3: ROC curves for the stress condition experiment (left curve - Stress Levels inferring and right curve - Fatigue Levels inferring)

Table 5.4: AUROC for stress and fatigue classification, in each experiment

	Induced Stress Test		**Induced Fatigue Test**	
	Stress	Fatigue	Stress	Fatigue
AUROC	8296	8606	7560	8519

(ability to determine the non-stress/fatigue cases correctly) of the stress and fatigue classifier were calculated, using Equations 5.3, 5.4 and 5.5. The obtained results are shown in Table 5.5.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{5.3}$$

$$Sensitivity = \frac{TP}{TP+FN} \tag{5.4}$$

$$Specificity = \frac{TN}{TN+FP} \tag{5.5}$$

Table 5.5: Accuracy, Sensitivity and Specificity for the Stress and Fatigue detections

	Accuracy	***Specificity***	***Sensitivity***
Stress Classification	77,6%	67,6%	88,9%
Fatigue Classification	70,0%	60,0%	84,0%

5.2.2.2 Position Monitoring & Motion Tracking

In the Position Monitoring & Motion Tracking evaluation 3600 area values were generated from the tests. Once again, in order to evaluate the data generated from the tests, not only the area values were retrieved to a text file, as also the SYSTEC-FOF Sensor Cloud Historical Data Function was used, sending the areas from the Kinect's SELCOMP to the Cloud. The metrics defined to analyze the accuracy of this system *TAD* and *FAD* were used to calculate the *TADR* and *FADR* as specified in Equations 5.6 and 5.7.

$$TADR = \frac{TAD}{TAD + FAD} \times 100 \tag{5.6}$$

$$FADR = \frac{FAD}{TAD + FAD} \times 100 \tag{5.7}$$

The *TAD*, *FAD*, *FADR* and *TADR* parameters were calculated for all areas in general and for each in particular. This allows to analyze the overall system accuracy and determine if any particular area generated more false detections than the others. The values obtained are presented in Table 5.6.

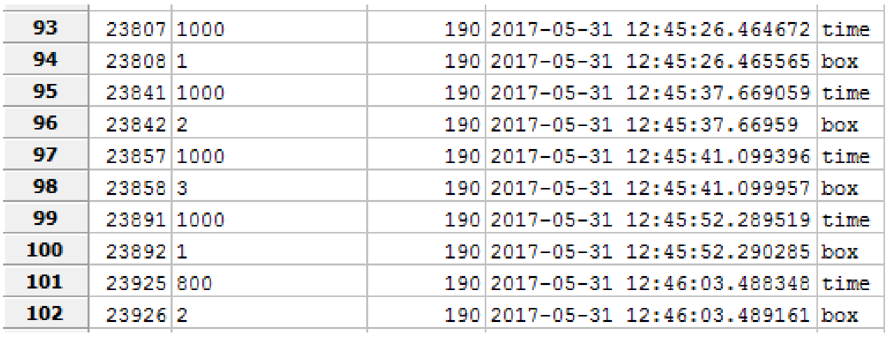
Table 5.6: Position Tracking evaluation parameters

	Area 0	***Area 1***	***Area 2***	***Area 3***	**Total**
TAD	559	511	497	484	2051
FAD	150	199	241	259	849
TADR	78,84%	71,97%	67,34%	65,14%	70,72%
FADR	21,16%	28,03%	32,66%	34,86%	29,28%

5.2.2.3 Use Case Scenario

In the Use Case Scenario evaluation the values returned from the MAPE-k (box and time) were stored in the Cloud database and retrieved after each test (Figure 5.4). In Tables 5.7 and 5.8 the

MET values are shown for subject A and B, respectively, for each box and adaptation combination, while in Table 5.9 the *METV* values for each subject and each stress/fatigue combination are presented. The *METV* of the *ij* stress/fatigue combination, was calculated using Equation 5.8.



93	23807	1000	190	2017-05-31 12:45:26.464672	time
94	23808	1	190	2017-05-31 12:45:26.465565	box
95	23841	1000	190	2017-05-31 12:45:37.669059	time
96	23842	2	190	2017-05-31 12:45:37.66959	box
97	23857	1000	190	2017-05-31 12:45:41.099396	time
98	23858	3	190	2017-05-31 12:45:41.099957	box
99	23891	1000	190	2017-05-31 12:45:52.289519	time
100	23892	1	190	2017-05-31 12:45:52.290285	box
101	23925	800	190	2017-05-31 12:46:03.488348	time
102	23926	2	190	2017-05-31 12:46:03.489161	box

Figure 5.4: Box and time values returned from MAPE-k

$$VMET_{ij} = \frac{MET_{ij} - MET_{\text{Low/Low}}}{MET_{\text{Low/Low}}} \times 100 \tag{5.8}$$

Table 5.7: MET values in milliseconds for each box and adaptation combination for Subject A

Subject A	***Box 1***	***Box 2***	***Box 3***	**Total**
Low/Low	600	666,7	600	622,2
Medium/Medium	1000	1000	933,3	977,8
High/High	1533,3	1533,3	1466,7	1511,1
Medium/High	1266,7	1133,3	1333,3	1244,4
High/Low	1133,3	1111,1	1155,6	1133,3
Medium/Low	888,9	933,3	866,7	896,3

Table 5.8: MET values in milliseconds for each box and adaptation combination for Subject B

Subject B	***Box 1***	***Box 2***	***Box 3***	**Total**
Low/Low	644,4	666,7	600	637
Medium/Medium	881,5	933,3	866,7	925,9
High/High	1177,8	1155,6	1133,3	1155,6
Medium/High	1111,1	1133,3	1133,3	1125,911
High/Low	1066,7	1088,9	1044,4	1066,7
Medium/Low	866,7	1000	911,1	893,8

Table 5.9: Ideal METV and METV for each subject

	METV Subject A	*METV Subject B*	*Ideal METV*
Low/Low	0%	0%	0%
Medium/Medium	57%	40%	67%
High/High	143%	81%	167%
Medium/High	100%	77%	133%
High/Low	82%	67%	100%
Medium/Low	44%	45%	33%

5.2.3 Discussion of Results

5.2.3.1 Stress and Fatigue Classifier

In the Stress and Fatigue Classifier evaluation, regarding the results presented in Table 5.2 from the ROC analysis for each subject, one can conclude that subject A has globally better results than Subject B, possibly due to the fact that different categories of people may respond differently to stress and fatigue stimuli, so the membership functions used in the fuzzy classifier could be more fit to the biosignals of Subject A. The overall ROC analysis shown in Table 5.3 demonstrates that the induced stress experiment obtained a TP rate of 91% for the stress detection and 75% for fatigue, while the TN rate was of 50% for stress and 89% for fatigue. This 50% TNR can be explained by the fact that humans are poor assessors of their own stress, whereas fatigue is more easily classified, so the subject could not perceive his own stress condition, since the value of the TNR increases in other test conditions. The AUROC values are presented in Table 5.4, where all the areas are above 7500, which demonstrates good reliability for the classification method. In terms of the accuracy, specificity and sensitivity values showed in Table 5.5, better results were obtained for stress classification than for fatigue, obtaining 77,6% accuracy, 67,6% specificity and 88,9% sensitivity for stress classification, while fatigue classification revealed 70% accuracy, 60% specificity, and 84% sensitivity.

While testing the Stress and Fatigue Classifier, most of the false positives occurred due to peaks in the EDA signal, which led to mistakes in the SCR. This can be explained by occasional movement of the left hand of the subject, where the electrodes were placed. Also, the fact that the fuzzy classifier memberships were obtained from data from a single subject, brought more accurate results to the inputs of subject A than subject B, since biosignals and its features highly variate with individual characteristics (gender, age, weight, among other factors). Moreover, given

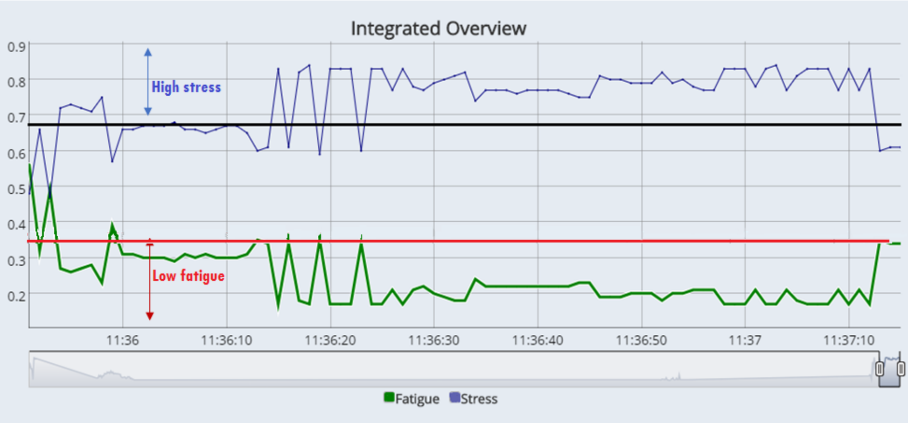


Figure 5.5: Graphic with Stress and Fatigue Levels from the Sensor Cloud Historical Data

that the stress and fatigue conditions were artificially induced, it was very hard to maintain those conditions throughout the tests. In Figure 5.5 these limitations can be observed, showing peaks occurred in stress and fatigue levels, that led to incorrect classifications in some moments.

5.2.3.2 Position Monitoring & Motion Tracking

In the Position Monitoring & Motion Tracking evaluation from the values acquired nearly 500 were discarded in order not to wrongly consider them false detection, as they were transition values that occur when the subject changes from one area to another, having to cross some other in between (like showed in the right side graphic in Figure 5.6, when crossing from area 2 to area 3 - time 8:51:10). Figure 5.6 presents two examples of the graphics provided by the Sensor Cloud Historical Data where some irregularities in the areas detection are demonstrated right after 9:10 in the left graphic, and after 8:51:30, when detecting area 1, in the right side graphic.

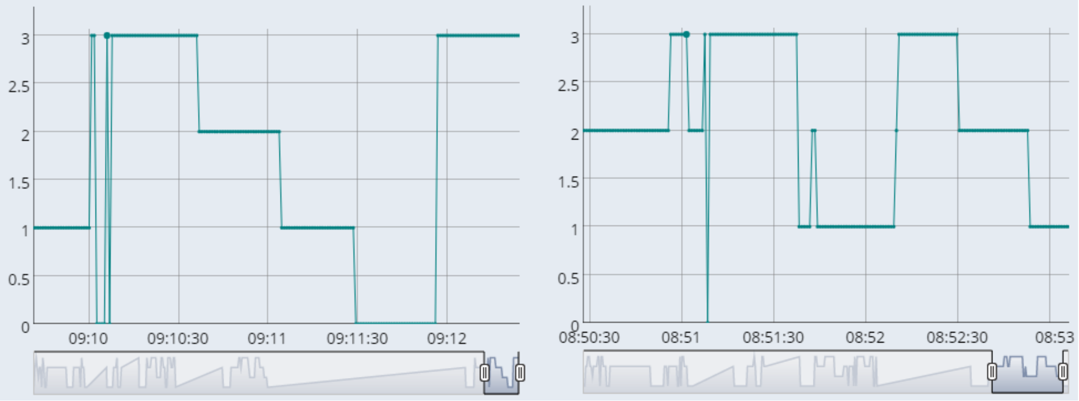


Figure 5.6: Graphics with the areas returned in two of the position tracking tests from the Sensor Cloud Historical Data (x-axis:time, y-axis:area)

From the general parameters calculated showed in Table 5.6, it can be concluded that the tracking system has an overall satisfactory performance, given that its *TADR* is almost 71%, while the percentage of areas incorrectly detected is of 29%. The *TADR* for the area 0 is clearly higher than for the other areas, this fact can be due to the dimension of that area being much greater in comparison, not being as affected to the discrepancies stemming from the right-hand tracking noise (see Subsection 4.2.1). In contrast, area 3 has the highest *FADR*, which is explained due to its position right behind the robotic arm, that obstructs Kinect's vision and sometimes causes it "loose" the right-hand position.

While performing the tests some limitations were perceived that demanded special conditions in order to achieve the position tracking. First, it was concluded that the individual needed to be sited in the exact same position since the tracking system has the coordinates for each area defined in the code, so if the person moved considerably from the spot used to initially "calibrate" the Kinect, the system would not correctly track if the right hand was in a determined area. Another limitation was the hand movement's quickness due to the latency existing in the body tracking. In general, the results are satisfactory but, for industrial applications, the system needs improvements and more testing scenarios and conditions.

5.2.3.3 Use Case Scenario

In the Use Case Scenario testing, when comparing the MET values obtained for each subject, presented in Tables 5.7 and 5.8, it can be concluded that in both cases the system adapted to the individual's state, demonstrating superior MET values in higher fatigue and stress levels (High/High and Medium/High combinations), increasing the robotic arm's time of executing its tasks, and consequently reducing the speed of each task. On the other hand, the $METV$ allows perceiving the level of adaptation, when compared to the normal execution conditions of the co-operative process. As it can be observed in Table 5.9, in the Subject A tests there is at least a 40% difference between the $METV$ of almost every stress/fatigue classification combinations, which indicates that the adaptation is successful and easily observed during the process. The only exception occurs between the Medium/Low and Medium/Medium cases and it can be explained due to the difficulty in maintaining both stress and fatigue at a medium level. When it comes to Subject

B, the $METV$ values are lower but still, there is a difference of almost 10% between classification combinations (again only between the Medium/Low and Medium/Medium that difference is lower). This difference between the two subjects is due to better stress and fatigue classification in the subject A as explained in the earlier.

Finally, when observing the difference between the ideal $METV$ values and the ones obtained, a discrepancy is clear. Since the use case scenario involves the fatigue/stress monitoring combined with the position monitoring, it is natural that the values obtained are less accurate than the optimal, as there is inaccuracy associated to those monitoring systems. Also, maintaining a specific induced combination of stress and fatigue in an individual is very hard, especially maintain it throughout the execution of the tests. Nonetheless $METV$ values obtained are satisfactory and in the case of the Subject A close to the ideal.

Chapter 6

Conclusions & Future Work

This chapter is reserved for final conclusions of the work done through this semester and possibilities of future work in aspects of optimizations and additional functionalities.

6.1 Conclusions

This dissertation focused essentially on the implementation of a CHMI system: a robotic arm, Kinect, and BITalino virtualized in a CPPS, with the main objective of adapting the robot to the operator's emotional state in terms of stress and fatigue, and also to his task order preference when in the cooperative process.

The ECG and EDA sensors were used, acquiring the signals with BITalino, and later processing them, making the acquisition more robust to obtain the metrics that allowed to infer stress and fatigue levels. This acquisition showed several limitations, as the data was very sensitive to the manner of sensor placement and sudden body motion, causing interference in the signals and consequently erroneous values in the metrics. Also, the signals suffered from day-to-day and subject variability, so the definition of processing methods that surpassed these variations and encompassed all types of signals and people were very hard to attain.

The fuzzy classifier used allowed to retrieve the wanted levels in a very intuitive way, but the development revealed itself very problematic, considering that for the memberships construction a set of reliable metrics when in stress and fatigue had to be gathered, and gathering physiological data related to mental stress and fatigue in a human being is not an easy task since it is difficult to simulate and sustain. The tests performed on this detection system revealed an accuracy of 77,6% for stress classification and 70% for fatigue. This lower result for fatigue can be explained with a better signals' correlation to stress than to fatigue (most human fatigue detectors use EEG signals). It is also important to refer that when comparing the results of different subjects the accuracy also varies due to the earlier explained variation in the biosignals from individual to individual.

For the Position monitoring & Motion tracking, the Kinect human tracking functions helped to facilitate the process. After some initial issues with the mapping of the joints, the detection of the right hand was relatively easy, only having to compare its position with the predefined areas.

Although this system shows overall satisfactory results (70,7% of true area detection rate) and allows an easy adaptation to other industrial processes, it has many limitations: from the user's restrictions in movements and positions and only suitable for right-handed individuals, to the light conditions and necessity of initial "calibration".

The MAPE-K implemented used the data acquired from the monitoring systems to determine if there was a need for adaptation, posteriorly sending commands with the adaptation parameters (execution time and box) to the robotic arm if a change was needed. This implementation was facilitated by the use of the Sensor Cloud that allowed to retrieve the data from the several sensors.

Finally by creating Smart Components of all physical devices existing in the system, and connecting them to a network, i.e., by creating a CPPS, the overall CHMI process became more efficient and flexible in terms of the design and operation of the robotic arm and complete process, and facilitated reconfigurations to respond to changes in its surrounding environment in real time, or in this dissertation case, to the changes in the operator.

With the implementation of the CPPS (including a Sensor Cloud integration) for the CHMI solution, much advantages emerged, like allowing to easily change HMI process if intended, facilitating environment/operator's monitoring and posterior visualization of data through the cloud, decreasing the response time and the cost of building and operating the system when compared to the usual automation solutions. It is important to emphasize that although all constituent parts of this system were implemented for this particular case, with the CPPS this solution could easily be adapted to other scenarios with different physical devices and different adaptation parameters.

6.2 Future work

Regarding all the work done in the implementation of this CHMI solution, there are still some aspects that can be improved and tested.

First, in order to allow more freedom of movement when performing the signals acquisition, the BITalino board should be transformed into a wearable that would not allow the electrodes to disconnect so easily. Also, the fuzzy classifier could be trained, being able to provide optimum outputs by adapting its rules and membership functions to the training data, potentially increasing the accuracy of the stress/fatigue classifier for several individuals.

With regard to the monitoring position system, the use of image processing to detect the boxes, or white markers to delimit the areas, also as making the system adaptable for both left-handed and right-hand individuals, would not only improve the precision of the detection but also increase the adaptability of the system.

When it comes to the adaptation to the human operator, other scenarios could be tested, like compensating the operator's decrease in productivity instead of trying to facilitate his work conditions. Additionally, a learning module could be implemented in the robotic arm, ceasing to be purely reactive to the surrounding environment and allowing it to perceive and adapt its actions based on acquired knowledge.

Finally, in order to specifically evaluate the impact of the CPPS in the overall solution, a comparison with an identical CHMI process that did not use CPS would have been very enlightening in what extent did actually the CPS improve the process, in terms of adaptation, latency and execution times.

Appendix A

BITalino Board Kit Data Sheet

BITalino Board
Kit Data Sheet

SPECIFICATIONS

> **Sampling Rate:** 1, 10, 100 or 1000Hz
> **Analog Ports:** 4 in (10-bit) + 2 in (6-bit)
> **Digital Ports:** 4 in (1-bit) + 4 out (1-bit)
> **Communication:** Bluetooth 2.0+EDR
> **Range:** up to ~10m (in line of sight)
> **Sensors:** EMG; ECG; EDA; ACC; LUX
> **Actuators:** LED
> **Size:** 105x60x6mm
> **Battery:** 500mA 3.7V LiPo (rechargeable)
> **Consumption:** ~65mA (everything active)
> **Accessories:** 3-lead cable (EMG/ECG);
2-lead cable (ECG/EDA); 5 electrodes

FEATURES

> Perfect for biosignal exploration
> All-in-one system
> Snappable blocks
> Raw data acquisition
> On-board battery charger
> Easy-to-use
> Affordable

APPLICATIONS

> Psychophysiology
> Biomedical projects
> Computer science
> Electrical engineering
> Human-Computer Interaction
> Robotics & Cybernetics
> Physiology studies
> Biomechanics
> Biofeedback

GENERAL DESCRIPTION

Our signature BITalino Board kit has an all-in-one ready-to-use hardware design, with all the blocks pre-connected between them, making it perfect for biosignal exploration and lab activities. The kit has all the accessories needed to get started, namely the hardware modules, battery, cables, and electrodes. Along with our cross-platform OpenSignals software, it enables instant biosignal data visualization and recording.

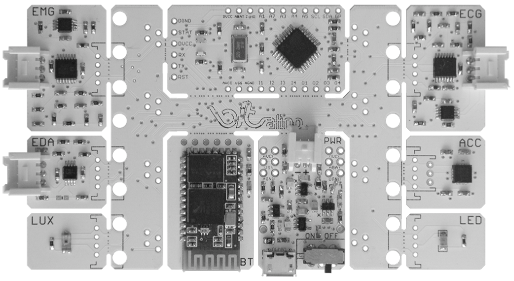


Fig. 1. All-in-one design ready to use out-of-the-box.

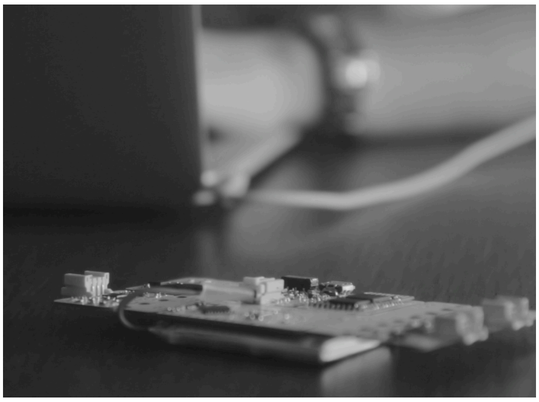


Fig. 2. The BITalino Board kit enables you to have your own personal biosignal acquitision system.

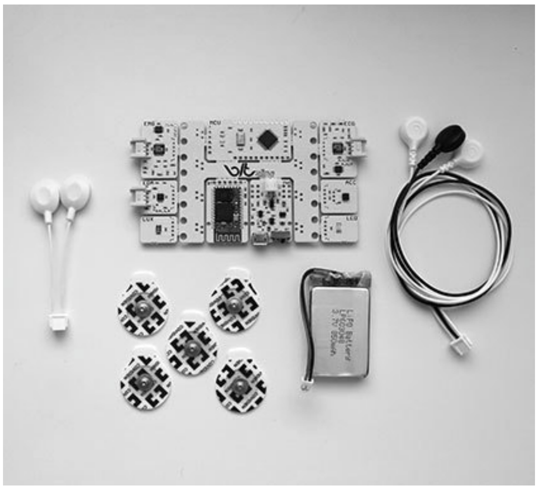


Fig. 3. Parts and accessories included in the kit.

PLUX – Wireless Biosignals, S.A.
Av. 5 de Outubro, n. 70 – 8.
1050-059 Lisbon, Portugal
bitalino@plux.info
http://bitalino.com/

REV A

This information is provided "as is," and we make no express or implied warranties whatsoever with respect to functionality, operability, use, fitness for a particular purpose, or infringement of rights. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

BEWARE: DIRECT OR INDIRECT COUPLING TO THE MAINS MAY RESULT IN SHOCKING HAZARD

Appendix B

Electrocardiography (ECG) Sensor Data Sheet

Electrocardiography (ECG) Sensor Data Sheet

ECG 151015

SPECIFICATIONS

> **Gain:** 1100
> **Range:** ±1.5mV (with VCC = 3.3V)
> **Bandwidth:** 0.5-40Hz
> **Consumption:** ~4mA
> **Input Impedance:** 100GOhm
> **CMRR:** 110dB
> **Electrodes:** 3 or 2 (virtual REF)

FEATURES

> Bipolar differential measurement
> Pre-conditioned analog output
> High signal-to-noise ratio
> Small form factor
> Raw data output
> Easy-to-use
> “On-the-person” and “off-the-person” use

APPLICATIONS

> Heart rate & heart rate variability
> Human-Computer Interaction
> Biometrics
> Affective computing
> Physiology studies
> Psychophysiology
> Biofeedback
> Biomedical devices prototyping

GENERAL DESCRIPTION

Heartbeats are triggered by bioelectrical signals of very low amplitude generated by a special set of cells in the heart (the SA node). Electrocardiography (ECG) enables the translation of these electrical signals into numerical values, enabling them to be used in a wide array of applications. Our sensor allow data acquisition not only at the chest (“on-the-person”), but also at the hand palms (“off-the-person”), and works both with pre-gelled and most types of dry electrodes. The bipolar configuration is ideal for low noise. Here are a few examples:
http://vimeo.com/98075534h
https://www.youtube.com/watch?v=hQ3BUBV-BBMh
https://www.youtube.com/watch?v=lKWM2TL1toM

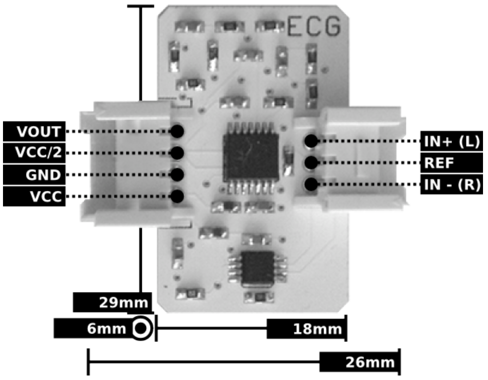


Fig. 1. Pin-out and physical dimensions.

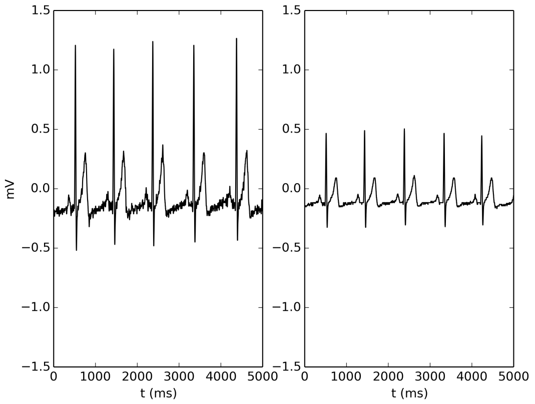


Fig. 2. Typical raw ECG data (acquired with BITalino) using 2 electrodes at the hands (left) and 3 electrodes at the chest (right).

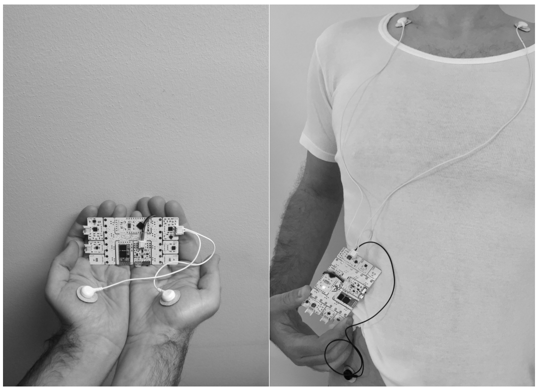


Fig. 3. Example of a 2 electrodes placement at the hands (left) and a 3 electrodes placement at the chest (right).

PLUX – Wireless Biosignals, S.A.
Av. 5 de Outubro, n. 70 – 8.
1050-059 Lisbon, Portugal
bitalino@plux.info
http://bitalino.com/

REV A

This information is provided "as is," and we make no express or implied warranties whatsoever with respect to functionality, operability, use, fitness for a particular purpose, or infringement of rights. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

BEWARE: DIRECT OR INDIRECT COUPLING TO THE MAINS MAY RESULT IN SHOCKING HAZARD

TRANSFER FUNCTION

[-1.5mV, 1.5mV]

$$ECG(V) = \frac{\left(\frac{ADC}{2^n} - \frac{1}{2}\right).VCC}{G_{ECG}}$$

$$ECG(mV) = ECG(V).1000$$

$VCC = 3.3V$ (operating voltage)
$G_{ECG} = 1100$ (sensor gain)

$ECG(V)$ – ECG value in Volt (V)
$ECG(mV)$ – ECG value in millivolt (mV)
ADC – Value sampled from the channel
n – Number of bits of the channel[1]

[1] The number of bits for each channel depends on the resolution of the Analog-to-Digital Converter (ADC); in BITalino the first four channels are sampled using 10-bit resolution ($n = 10$), while the last two are sampled using 6-bit ($n = 6$).

Appendix C

Electrodermal Activity (EDA) Sensor Data Sheet

Electrodermal Activity (EDA) Sensor Data Sheet

EDA 151015

SPECIFICATIONS

> **Gain:** 2
> **Range:** 0-1MOhm (with VCC = 3.3V)
> **Bandwidth:** 0-3Hz
> **Consumption:** ~2mA
> **Electrodes:** 2

FEATURES

> Skin resistance measurement
> Pre-conditioned analog output
> High signal-to-noise ratio
> Small form factor
> Raw data output
> Easy-to-use

APPLICATIONS

> Arousal detection
> Human-Computer Interaction
> Emotional cartography
> Affective computing
> Physiology studies
> Psychophysiology
> Relaxation biofeedback
> Biomedical devices prototyping

GENERAL DESCRIPTION

Sweat glands secretion is a process that allows our body to regulate its temperature, but it is also associated the sympathetic nervous system activity. Whenever we become aroused (e.g. nervous) or relaxed, that state is partially translated into the sweat production or inhibition at the glands on our hands palms and feet. This changes the resistance of our skin; Electrodermal Activity (EDA) monitoring enables the translation of these resistance changes into numerical values, allowing its use in a wide array of applications. Known uses of this sensor include emotional mapping, the polygraph test (aka lie detector), and also stress / relaxation biofeedback.

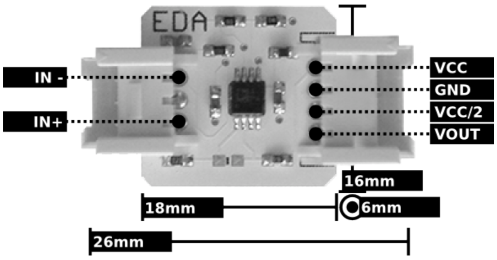


Fig. 1. Pin-out and physical dimensions.

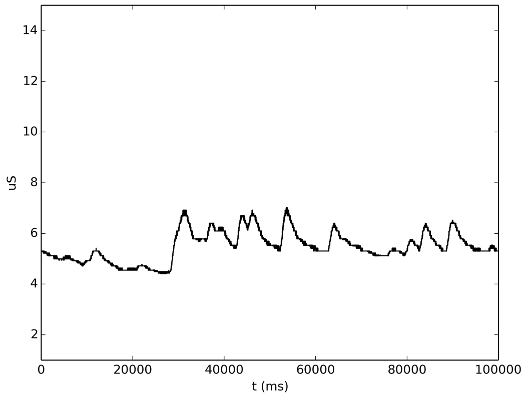


Fig. 2. Typical raw EDA data (acquired with BITalino).

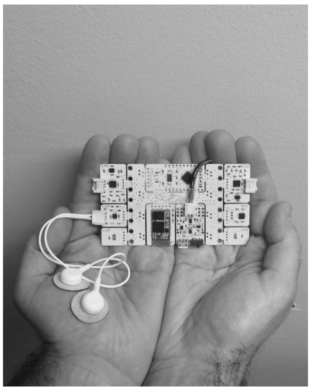


Fig. 3. Example electrode placement.

PLUX – Wireless Biosignals, S.A.
Av. 5 de Outubro, n. 70 – 8.
1050-059 Lisbon, Portugal
bitalino@plux.info
http://bitalino.com/

REV A

This information is provided "as is," and we make no express or implied warranties whatsoever with respect to functionality, operability, use, fitness for a particular purpose, or infringement of rights. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

BEWARE: DIRECT OR INDIRECT COUPLING TO THE MAINS MAY RESULT IN SHOCKING HAZARD

TRANSFER FUNCTION

$[1\mu S, \infty\mu S]$

$$R(MOhm) = 1 - \frac{ADC}{2^n}$$

$$EDA(\mu S) = \frac{1}{R(MOhm)}$$

$R(MOhm)$ – Sensor resistance value mega-Ohm ($MOhm$)
$EDA(\mu S)$ – EDA value in micro-Siemens (μS)
ADC – Value sampled from the channel
n – Number of bits of the channel[1]

[1] The number of bits for each channel depends on the resolution of the Analog-to-Digital Converter (ADC); in BITalino the first four channels are sampled using 10-bit resolution ($n = 10$), while the last two are sampled using 6-bit ($n = 6$).

Appendix D

Electromyography (EMG) Sensor Data Sheet

Electromyography (EMG) Sensor Data Sheet

EMG 151015

SPECIFICATIONS

> **Gain:** 1000
> **Range:** ±1.65mV (with VCC = 3.3V)
> **Bandwidth:** 10-400Hz
> **Consumption:** ~4mA
> **Input Impedance:** 100GOhm
> **CMRR:** 110dB

FEATURES

> Bipolar differential measurement
> Pre-conditioned analog output
> High signal-to-noise ratio
> Small form factor
> Raw data output
> Easy-to-use

APPLICATIONS

> Human-Computer Interaction
> Robotics & Cybernetics
> Physiology studies
> Psychophysiology
> Biomechanics
> Biofeedback
> Muscle reflex studies
> Nerve conduction measurement
> Biomedical devices prototyping

GENERAL DESCRIPTION

Muscle activation is triggered by bioelectrical signals of very low amplitude sent from motor control neurons on our brain to the muscle fibers. Electromyography (EMG) enables the translation of these electrical signals into numerical values, enabling them to be used in a wide array of applications. Our sensor is especially designed for surface EMG, and works both with pre-gelled and most types of dry electrodes. The bipolar configuration is ideal for low-noise data acquisition, and the raw data output enables it to be used for human-computer interaction and biomedical projects alike. Here are a few examples:

https://www.youtube.com/watch?v=pVAaFeym8TQ
https://www.youtube.com/watch?v=7Q4HC0vxFsc
http://www.physioplux.com/

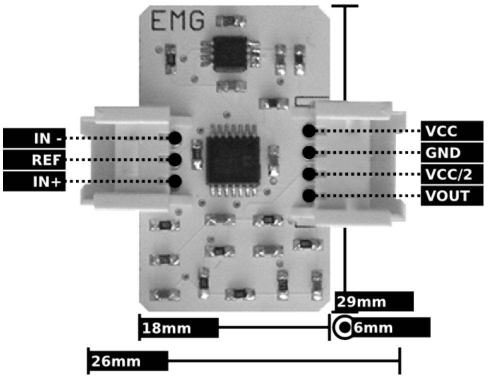


Fig. 1. Pin-out and physical dimensions.

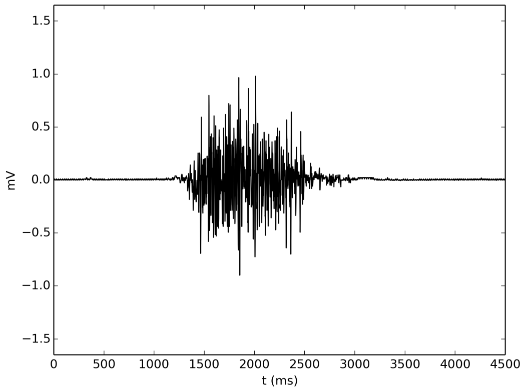


Fig. 2. Typical raw EMG data (acquired with BITalino).

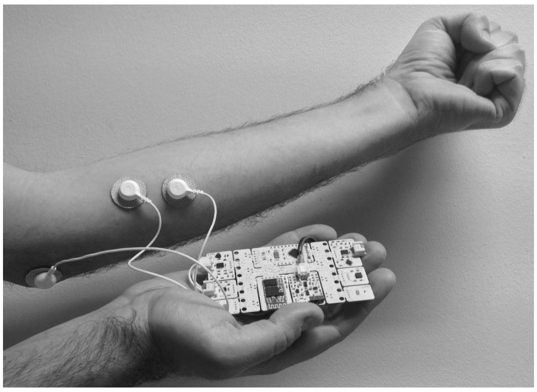


Fig. 3. Example electrode placement, with REF in a bone region (electrically neutral), and IN+ & IN- 20mm apart over the muscle belly (aligned with the muscle fibers).

PLUX – Wireless Biosignals, S.A.
Av. 5 de Outubro, n. 70 – 8.
1050-059 Lisbon, Portugal
bitalino@plux.info
http://bitalino.com/

REV A

This information is provided "as is," and we make no express or implied warranties whatsoever with respect to functionality, operability, use, fitness for a particular purpose, or infringement of rights. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

BEWARE: DIRECT OR INDIRECT COUPLING TO THE MAINS MAY RESULT IN SHOCKING HAZARD

TRANSFER FUNCTION

[-1.65mV, 1.65mV]

$$EMG(V) = \frac{\left(\frac{ADC}{2^n} - \frac{1}{2}\right).VCC}{G_{EMG}}$$

$$EMG(mV) = EMG(V).1000$$

$VCC = 3.3V$ (operating voltage)
$G_{EMG} = 1000$ (sensor gain)

$EMG(V)$ – EMG value in Volt (V)
$EMG(mV)$ – EMG value in millivolt (mV)
ADC – Value sampled from the channel
n – Number of bits of the channel[1]

[1] The number of bits for each channel depends on the resolution of the Analog-to-Digital Converter (ADC); in BITalino the first four channels are sampled using 10-bit resolution ($n = 10$), while the last two are sampled using 6-bit ($n = 6$).

Appendix E

Fuzzy Module Rules

1. IF *HR* is Low AND *IBI* is High AND *RATIO* is Low AND *SCR* not High THEN *Stress* is Low AND *Fatigue* is Low

2. IF *HR* is Low AND *IBI* is High AND *RATIO* is Medium AND *SCR* not High THEN *Stress* is Low AND *Fatigue* is Medium

3. IF *HR* is Low AND *IBI* is High AND *RATIO* is High AND *SCR* not High THEN *Stress* is Low AND *Fatigue* is Low

4. IF *HR* is Low AND *IBI* is High AND *RATIO* is High AND *SCR* High AND *SCL* High THEN *Stress* is Low AND *Fatigue* is Medium

5. IF *HR* is Low AND *IBI* is High AND *RATIO* is Medium AND *SCR* High AND *SCL* High THEN *Stress* is Low AND *Fatigue* is High

6. IF *HR* is Low AND *IBI* is High AND *RATIO* is Low AND *SCR* High AND *SCL* High THEN *Stress* is Low AND *Fatigue* is High

7. IF *HR* is Low AND *IBI* is High AND *RATIO* is High AND *SCR* High AND *SCL* Low THEN *Stress* is Low AND *Fatigue* is Medium

8. IF *HR* is Low AND *IBI* is High AND *RATIO* is Medium AND *SCR* High AND *SCL* Low THEN *Stress* is Low AND *Fatigue* is Medium

9. IF *HR* is Low AND *IBI* is High AND *RATIO* is Low AND *SCR* High AND *SCL* Low THEN *Stress* is Low AND *Fatigue* is High

10. IF *HR* is Medium AND *IBI* is Medium AND *RATIO* is Low AND *SCR* not High THEN *Stress* is low AND *Fatigue* is Medium

11. IF *HR* is Medium AND *IBI* is Medium AND *RATIO* is High AND *SCR* not High THEN *Stress* is Medium AND *Fatigue* is Low

12. IF HR is Medium AND IBI is Medium AND $RATIO$ is Medium AND SCR not High THEN $Stress$ is Medium AND $Fatigue$ is Medium
13. IF HR is Medium AND IBI is Medium AND $RATIO$ is High AND SCR High AND SCL High THEN $Stress$ is High AND $Fatigue$ is Low
14. IF HR is Medium AND IBI is Medium AND $RATIO$ is Medium AND SCR High AND SCL High THEN $Stress$ is Medium AND $Fatigue$ is Low
15. IF HR is Medium AND IBI is Medium AND $RATIO$ is Low AND SCR High AND SCL High THEN $Stress$ is Medium AND $Fatigue$ is Medium
16. IF HR is Medium AND IBI is Medium AND $RATIO$ is High AND SCR High AND SCL Low THEN $Stress$ is Medium AND $Fatigue$ is Low
17. IF HR is Medium AND IBI is Medium AND $RATIO$ is Medium AND SCR High AND SCL Low THEN $Stress$ is High AND $Fatigue$ is Low
18. IF HR is Medium AND IBI is Medium AND $RATIO$ is Low AND SCR High AND SCL Low THEN $Stress$ is Medium AND $Fatigue$ is Medium
19. IF HR is High AND IBI is Low AND $RATIO$ is Low AND SCR not High THEN $Stress$ is Medium AND $Fatigue$ is Low
20. IF HR is High AND IBI is Low AND $RATIO$ is High AND SCR not High THEN $Stress$ is High AND $Fatigue$ is Low
21. IF HR is High AND IBI is Low AND $RATIO$ is Medium AND SCR not High THEN $Stress$ is Medium AND $Fatigue$ is Low
22. IF HR is High AND IBI is Low AND $RATIO$ is High AND SCR High AND SCL High THEN $Stress$ is High AND $Fatigue$ is Low
23. IF HR is High AND IBI is Low AND $RATIO$ is Medium AND SCR High AND SCL High THEN $Stress$ is High AND $Fatigue$ is Low
24. IF HR is High AND IBI is Low AND $RATIO$ is Low AND SCR High AND SCL High THEN $Stress$ is High AND $Fatigue$ is Medium
25. IF HR is High AND IBI is Low AND $RATIO$ is High AND SCR High AND SCL Low THEN $Stress$ is High AND $Fatigue$ is Low
26. IF HR is High AND IBI is Low AND $RATIO$ is Medium AND SCR High AND SCL Low THEN $Stress$ is High AND $Fatigue$ is Low
27. IF HR is High AND IBI is Low AND $RATIO$ is Low AND SCR High AND SCL Low THEN $Stress$ is Medium AND $Fatigue$ is Low

References

[1] Jiafu Wan, Hu Cai, and Keliang Zhou. Industrie 4.0: enabling technologies. In *Intelligent Computing and Internet of Things (ICIT), 2014 International Conference on*, pages 135–140. IEEE, 2015.

[2] Didac Gil De La Iglesia and Danny Weyns. Mape-k formal templates to rigorously design behaviors for self-adaptive systems. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 10(3):15, 2015.

[3] Cornelia Setz, Bert Arnrich, Johannes Schumm, Roberto La Marca, Gerhard Tröster, and Ulrike Ehlert. Discriminating stress from cognitive load using a wearable eda device. *IEEE Transactions on information technology in biomedicine*, 14(2):410–417, 2010.

[4] Mobyen Uddin Ahmed, Shahina Begum, Peter Funk, and Ning Xiong. Fuzzy rule-based classification to build initial case library for case-based stress diagnosis. In *the proceedings of 9th International Conference on Artificial Intelligence and Applications (AIA)*, pages 225–230, 2009.

[5] Hugo Plácido da Silva, Ana Fred, and Raúl Martins. Biosignals for everyone. *IEEE Pervasive Computing*, 13(4):64–71, 2014.

[6] Microsoft Corporation. Human Interface Guidelines. *Microsoft Corporation*, page 70, 2012.

[7] H Gonzalez-Jorge, P Rodríguez-Gonzálvez, J Martínez-Sánchez, D González-Aguilera, P Arias, M Gesto, and L Díaz-Vilariño. Metrological comparison between kinect i and kinect ii sensors. *Measurement*, 70:21–26, 2015.

[8] SPARC. Robotics 2020 - Multi-Annual Roadmap For Robotics in Europe. *Horizon 2020 Call ICT-2016 (ICT-25 & ICT-26), Release B 03/12/2015*, 2015.

[9] Roberta Nelson Shea. Robot safety standard update. In *RIA meeting*, pages 20–23, 2013.

[10] D Gorecky, R Campos, and G Meixner. Seamless augmented reality support on the shopfloor based on cyber-physical-systems. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, 2012.

[11] Roger Moore. Presence: A human-inspired architecture for speech-based human-machine interaction. *IEEE Transactions on Computers*, 56(9), 2007.

[12] László Monostori. Cyber-physical production systems: Roots, expectations and r&d challenges. *Procedia CIRP*, 17:9–13, 2014.

[13] Jörg Krüger, Terje K Lien, and Alexander Verl. Cooperation of human and machines in assembly lines. *CIRP Annals-Manufacturing Technology*, 58(2):628–646, 2009.

[14] Panagiota Tsarouchi, Alexandros-Stereos Matthaiakis, Sotiris Makris, and George Chryssolouris. On a human-robot collaboration in an assembly cell. *International Journal of Computer Integrated Manufacturing*, 30(6):580–589, 2017.

[15] Xi Vincent Wang, Zsolt Kemény, József Váncza, and Lihui Wang. Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation. *CIRP Annals-Manufacturing Technology*, 2017.

[16] Paulo Leitão. Holonic rationale and self-organization on design of complex evolvable systems. In *International Conference on Industrial Applications of Holonic and Multi-Agent Systems*, pages 1–12. Springer, 2009.

[17] Malte Brettel, Niklas Friederichsen, Michael Keller, and Marius Rosenberg. How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective. *International Journal of Mechanical, Industrial Science and Engineering*, 8(1):37–44, 2014.

[18] Roland Rosen, Georg von Wichert, George Lo, and Kurt D Bettenhausen. About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine*, 48(3):567–572, 2015.

[19] José Barbosa, Paulo Leitão, Damien Trentesaux, Armando W Colombo, and Stamatis Karnouskos. Cross benefits from cyber-physical systems and intelligent products for future smart industries. In *Industrial Informatics (INDIN), 2016 IEEE 14th International Conference on*, pages 504–509. IEEE, 2016.

[20] Rui Pinto, Joao Reis, Ricardo Silva, Michael Peschl, and Gil Gonçalves. Smart sensing components in advanced manufacturing systems. *International Journal on Advances in Intelligent Systems*, 9(1&2):181–198, 2016.

[21] Luis Neto, João Reis, Ricardo Silva, and Gil Gonçalves. Sensor selcomp, a smart component for the industrial sensor cloud of the future. In *Industrial Technology (ICIT), 2017 IEEE International Conference on*, pages 1256–1261. IEEE, 2017.

[22] Kazi Masudul Alam and Abdulmotaleb El Saddik. C2ps: A digital twin architecture reference model for the cloud-based cyber-physical systems. *IEEE Access*, 5:2050–2062, 2017.

[23] Song Han, Miao Xie, Hsiao-Hwa Chen, and Yun Ling. Intrusion detection in cyber-physical systems: Techniques and challenges. *IEEE Systems Journal*, 8(4):1052–1062, 2014.

[24] Eric Ke Wang, Yunming Ye, Xiaofei Xu, Siu-Ming Yiu, Lucas Chi Kwong Hui, and Kam-Pui Chow. Security issues and challenges for cyber physical system. In *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, pages 733–738. IEEE Computer Society, 2010.

[25] Birgit Vogel-Heuser, Christian Diedrich, Dorothea Pantförder, and Peter Göhner. Coupling heterogeneous production systems by a multi-agent based cyber-physical production system. In *Industrial Informatics (INDIN), 2014 12th IEEE International Conference on*, pages 713–719. IEEE, 2014.

[26] Jay Lee, Behrad Bagheri, and Hung-An Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18–23, 2015.

[27] IBM. An architectural blueprint for autonomic computing. Technical report, 2005. URL: http://www-03.ibm.com/autonomic/pdfs/ACBlueprintWhitePaperV7.pdf.

[28] Tom Cox. *Stress research and stress management: Putting theory to work*, volume 61. HSE Books Sudbury, 1993.

[29] Chunlin Zhao, Min Zhao, Jianpin Liu, and Chongxun Zheng. Electroencephalogram and electrocardiograph assessment of mental fatigue in a driving simulator. *Accident Analysis & Prevention*, 45:83–90, 2012.

[30] Sandy Smith. Severe Impact of Fatigue in the Workplace Examined, 2002. URL: http://ehstoday.com/news/ehs{_}imp{_}35340.

[31] Josh Cable. Study: Workplace Fatigue Common, Costly, 2007. URL: http://ehstoday.com/news/ehs{_}imp{_}44448.

[32] Pramila Rani, Jared Sims, Robert Brackin, and Nilanjan Sarkar. Online stress detection using psychophysiological signals for implicit human-robot cooperation. *Robotica*, 20(06):673–685, 2002.

[33] Tanja GM Vrijkotte, Lorenz JP Van Doornen, and Eco JC De Geus. Effects of work stress on ambulatory blood pressure, heart rate, and heart rate variability. *Hypertension*, 35(4):880–886, 2000.

[34] Saroj KL Lal and Ashley Craig. A critical review of the psychophysiology of driver fatigue. *Biological psychology*, 55(3):173–194, 2001.

[35] Mohit Kumar, Matthias Weippert, Reinhard Vilbrandt, Steffi Kreuzfeld, and Regina Stoll. Fuzzy evaluation of heart rate signals for mental stress assessment. *IEEE transactions on fuzzy systems*, 15(5):791–808, 2007.

[36] Kil-sang Yoo and Won-hyung Lee. Mental stress assessment based on pulse photoplethysmography. In *Consumer Electronics (ISCE), 2011 IEEE 15th International Symposium on*, pages 323–326. IEEE, 2011.

[37] F Mokhayeri, MR Akbarzadeh-T, and S Toosizadeh. Mental stress detection using physiological signals based on soft computing techniques. In *Biomedical Engineering (ICBME), 2011 18th Iranian Conference of*, pages 232–237. IEEE, 2011.

[38] L Xia, J Wang, F Liang, W Li, J Guo, and Q Deng. Mental fatigue assessment based on physiological signals. *Nan fang yi ke da xue xue bao= Journal of Southern Medical University*, 32(6):870–873, 2012.

[39] Alberto de Santos Sierra, Carmen Sánchez Ávila, Javier Guerra Casanova, and Gonzalo Bailador del Pozo. A stress-detection system based on physiological signals and fuzzy logic. *IEEE Transactions on Industrial Electronics*, 58(10):4857–4865, 2011.

[40] Jennifer A Healey and Rosalind W Picard. Detecting stress during real-world driving tasks using physiological sensors. *IEEE Transactions on intelligent transportation systems*, 6(2):156–166, 2005.

[41] M Patel, SKL Lal, Diarmuid Kavanagh, and Peter Rossiter. Applying neural network analysis on heart rate variability data to assess driver fatigue. *Expert systems with Applications*, 38(6):7235–7242, 2011.

[42] Ulf Lundberg, Roland Kadefors, Bo Melin, Gunnar Palmerud, Peter Hassmén, Margareta Engström, and Ingela Elfsberg Dohns. Psychophysiological stress and emg activity of the trapezius muscle. *International journal of behavioral medicine*, 1(4):354–370, 1994.

[43] Alejandro Riera, Aureli Soria-Frisch, Anton Albajes-Eizagirre, Pietro Cipresso, Carles Grau, Stephen Dunne, Giulio Ruffini, and S aStarlab Barcelona. Electro-physiological data fusion for stress detection. *Studies in health technology and informatics*, 181:228–32, 2012.

[44] Dag Rissén, Bo Melin, Leif Sandsjö, Ingela Dohns, and Ulf Lundberg. Surface emg and psychophysiological stress reactions in women during repetitive work. *European journal of applied physiology*, 83(2-3):215–222, 2000.

[45] Qiang Ji, Zhiwei Zhu, and Peilin Lan. Real-time nonintrusive monitoring and prediction of driver fatigue. *IEEE transactions on vehicular technology*, 53(4):1052–1068, 2004.

[46] Soichiro Matsushita, Ayumi Shiba, and Kan Nagashima. A wearable fatigue monitoring system: application of human-computer interaction evaluation. In *Proceedings of the 7th Australasian User interface conference-Volume 50*, pages 161–164. Australian Computer Society, Inc., 2006.

[47] Feng-Tso Sun, Cynthia Kuo, Heng-Tze Cheng, Senaka Buthpitiya, Patricia Collins, and Martin Griss. Activity-aware mental stress detection using physiological sensors. In *International Conference on Mobile Computing, Applications, and Services*, pages 211–230. Springer, 2010.

[48] Nandita Sharma and Tom Gedeon. Objective measures, sensors and computational techniques for stress recognition and classification: A survey. *Computer methods and programs in biomedicine*, 108(3):1287–1301, 2012.

[49] Armando Barreto, Jing Zhai, and Malek Adjouadi. Non-intrusive physiological monitoring for automated stress detection in human-computer interaction. In *International Workshop on Human-Computer Interaction*, pages 29–38. Springer, 2007.

[50] Jing Zhai and Armando Barreto. Stress recognition using non-invasive technology. In *FLAIRS Conference*, pages 395–401, 2006.

[51] Wenhui Liao, Weihong Zhang, Zhiwei Zhu, and Qiang Ji. A real-time human stress monitoring system using dynamic bayesian network. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pages 70–70. IEEE, 2005.

[52] Youngjae Kim, Youmin Kim, and Minsoo Hahn. Detecting driver fatigue based on the driver's response pattern and the front view environment of an automobile. In *Universal Communication, 2008. ISUC'08. Second International Symposium on*, pages 237–240. IEEE, 2008.

[53] Kai-Quan Shen, Chong-Jin Ong, Xiao-Ping Li, Zheng Hui, and Einar PV Wilder-Smith. A feature selection method for multilevel mental fatigue eeg classification. *IEEE Transactions on Biomedical Engineering*, 54(7):1231–1237, 2007.

[54] Kai-Quan Shen, Xiao-Ping Li, Chong-Jin Ong, Shi-Yun Shao, and Einar PV Wilder-Smith. Eeg-based mental fatigue measurement using multi-class support vector machines with confidence estimate. *Clinical Neurophysiology*, 119(7):1524–1533, 2008.

[55] Norizam Sulaiman, Mohd Nasir Taib, Sahrim Lias, Zunairah Hj Murat, Siti Armiza Mohd Aris, and Noor Hayatee Abdul Hamid. Eeg-based stress features using spectral centroids technique and k-nearest neighbor classifier. In *Computer Modelling and Simulation (UKSim), 2011 UkSim 13th International Conference on*, pages 69–74. IEEE, 2011.

[56] Jeen-Shing Wang, Che-Wei Lin, and Ya-Ting C Yang. A k-nearest-neighbor classifier with heart rate variability feature-based transformation algorithm for driving stress recognition. *Neurocomputing*, 116:136–143, 2013.

[57] Mohit Kumar, Dagmar Arndt, Steffi Kreuzfeld, Kerstin Thurow, Norbert Stoll, and Regina Stoll. Fuzzy techniques for subjective workload-score modeling under uncertainties. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(6):1449–1464, 2008.

[58] Tayyaba Azim, M Arfan Jaffar, and Anwar M Mirza. Fully automated real time fatigue detection of drivers through fuzzy expert systems. *Applied Soft Computing*, 18:25–38, 2014.

[59] Weiqing Jin et al. Fuzzy classification based on fuzzy association rule mining. 2005.

[60] M Jordan, J Kleinberg, and B Schölkopf. Information science and statistics. *Springer, Cambridge, UK*, 2006.

[61] Chonghua Wang. A study of membership functions on mamdani-type fuzzy inference system for industrial decision-making. 2015.

[62] JM Morales, Carolina Díaz-Piedra, Leandro Luigi Di Stasi, S Romero, et al. Low-cost remote monitoring of biomedical signals. In *International Work-Conference on the Interplay Between Natural and Artificial Computation*, pages 288–295. Springer, 2015.

[63] Maurizio Garbarino, Matteo Lai, Dan Bender, Rosalind W Picard, and Simone Tognetti. Empatica e3—a wearable wireless multi-sensor device for real-time computerized biofeedback and data acquisition. In *Wireless Mobile Communication and Healthcare (Mobihealth), 2014 EAI 4th International Conference on*, pages 39–42. IEEE, 2014.

[64] Shalom Greene, Himanshu Thapliyal, and Allison Caban-Holt. A survey of affective computing for stress detection: Evaluating technologies in stress detection for better health. *IEEE Consumer Electronics Magazine*, 5(4):44–56, 2016.

[65] John Patrick Pullen. Whoop Fitness Band: Why Athletes Love It | Time.com, 2017. URL: http://time.com/4744459/whoop-strap-fitness-tracker-band/.

[66] CA Russell, JA Caldwell, D Arand, L Myers, P Wubbels, and H Downs. Validation of the fatigue science readiband actigraph and associated sleep/wake classification algorithms. *Archinoetics, LLC*, 2000.

[67] Hilal Abbood, Waleed Al-Nuaimy, Ali Al-Ataby, Sameh A Salem, and Hamzah S AlZubi. Prediction of driver fatigue: Approaches and open challenges. In *Computational Intelligence (UKCI), 2014 14th UK Workshop on*, pages 1–6. IEEE, 2014.

[68] Mitchell Gelpi, Chris Goode, and Olumayokun Odukale. NONMARKET ANALYSIS FOR MAVEN. Technical report, 2016.

[69] Svein Petter Gjøby. *Extensible data acquisition tool for Android.* PhD thesis, University of Olso, 2016.

[70] S Zennaro, Matteo Munaro, Simone Milani, Pietro Zanuttigh, A Bernardi, Stefano Ghidoni, and Emanuele Menegatti. Performance evaluation of the 1st and 2nd generation kinect for multimedia applications. In *Multimedia and Expo (ICME), 2015 IEEE International Conference on*, pages 1–6. IEEE, 2015.

[71] Oliver Wasenmüller and Didier Stricker. Comparison of kinect v1 and v2 depth images in terms of accuracy and precision. In *Asian Conference on Computer Vision Workshop (ACCV workshop), Springer*, 2016.

[72] Andrea Němcová, Lucie Maršánová, and Radovan Smíšek. Recommendations for ecg acquisition using bitalino.

[73] Kanika Tayal and Vijay Gill. Denoising of ECG Signal Using Savitzky Golay Filters. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO Certified Organization)*, 3297(4), 2007. URL: www.ijircce.com.

[74] Yong Lian and Jiang Hong Yu. The reduction of noises in ecg signal using a frequency response masking based fir filter. In *Biomedical Circuits and Systems, 2004 IEEE International Workshop on*, pages S2–4. IEEE, 2004.

[75] Maurice Bellanger. Digital processing of signals: theory and practice. *New York, Wiley-Interscience, 1984, 395 p. Translation.*, 1984.

[76] Guohua Lu, John-Stuart Brittain, Peter Holland, John Yianni, Alexander L Green, John F Stein, Tipu Z Aziz, and Shouyan Wang. Removing ecg noise from surface emg signals using adaptive filtering. *Neuroscience letters*, 462(1):14–19, 2009.

[77] Srija Unnikrishnan, Sunil Surve, and Deepak Bhoir. *Advances in Computing, Communication, and Control: Third International Conference, ICAC3 2013, Mumbai, India, January 18-19, 2013, Proceedings*, volume 361. Springer, 2013.

[78] Ko Keun Kim, Jung Soo Kim, Yong Gyu Lim, and Kwang Suk Park. The effect of missing rr-interval data on heart rate variability analysis in the frequency domain. *Physiological measurement*, 30(10):1039, 2009.

[79] George B. Moody. Frequency Domain Measures. URL: https://www.physionet.org/events/hrv-2006/moody-1.html.

[80] Kyung Hwan Kim, Seok Won Bang, and Sang Ryong Kim. Emotion recognition system using short-term monitoring of physiological signals. *Medical and biological engineering and computing*, 42(3):419–427, 2004.

[81] Gunilla Krantz, Mikael Forsman, and Ulf Lundberg. Consistency in physiological stress responses and electromyographic activity during induced stress exposure in women and men. *Integrative physiological and behavioral science*, 39(2):105–118, 2004.

[82] PLUX - Wireless Signals. *Electromyography (EMG) Sensor Data Sheet*, 2015.

[83] Lawrence M Schleifer, Thomas W Spalding, Scott E Kerick, Jeffrey R Cram, Ronald Ley, and Bradley D Hatfield. Mental stress and trapezius muscle activation under psychomotor challenge: a focus on emg gaps during computer work. *Psychophysiology*, 45(3):356–365, 2008.

[84] Jacqueline Wijsman, Bernard Grundlehner, Julien Penders, and Hermie Hermens. Trapezius muscle emg as predictor of mental stress. In *Wireless Health 2010*, pages 155–163. ACM, 2010.

[85] Janessa DM Drake and Jack P Callaghan. Elimination of electrocardiogram contamination from electromyogram signals: an evaluation of currently used removal techniques. *Journal of electromyography and kinesiology*, 16(2):175–187, 2006.

[86] Jarrett Webb and James Ashley. *Beginning Kinect Programming with the Microsoft Kinect SDK*. Apress, 2012.

[87] Abhijit Jana. *Kinect for Windows SDK Programming Guide*. 2012. URL: http://www.doc.flash-robotics.com/download/doc/KinectforWindowsSDKProgrammingGuide.pdf.

[88] Francisco José Macedo Fernandes. *Sit-To-Stand movement analysis using the Kinect platform*. PhD thesis, PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2013.

[89] Mark A Livingston, Jay Sebastian, Zhuming Ai, and Jonathan W Decker. Performance measurements for the microsoft kinect skeleton. In *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE*, pages 119–120. IEEE, 2012.

[90] Luis Neto, João Reis, Diana Guimarães, and Gil Gonçalves. Sensor cloud: Smartcomponent framework for reconfigurable diagnostics in intelligent manufacturing environments. In *Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on*, pages 1706–1711. IEEE, 2015.

[91] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.