Politecnico di Torino

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation
Doctoral Program in Computer Engineering (36^{th}cycle)

Multivariate Analysis in Research and Industrial Environments

By

Antonio Costantino Marceddu

******

Supervisor:
Prof. Bartolomeo Montrucchio

Doctoral Examination Committee:
Prof. Claudio Fornaro, Referee, Università Telematica Internazionale UNINETTUNO
Prof. Giancarlo Iannizzotto, Referee, Università degli Studi di Messina
Prof. Maja Matijasevic, University of Zagreb
Prof. Sorin-Aurel Moraru, Transilvania University of Brasov
Prof. Maurizio Rebaudengo, Politecnico di Torino

Politecnico di Torino
2024

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Antonio Costantino Marceddu
2024

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

I would like to dedicate this thesis
to my parents Graziano and Silvia,
to my sister Valentina,
and to my girlfriend Sandra.

Acknowledgements

I would like to acknowledge my supervisor, Prof. Bartolomeo Montrucchio, for overseeing my research work, and all my colleagues, the Master's thesis students, and the people with whom I collaborated during my Ph.D. These people are, in strict alphabetical order: Dr. Alessandro Aimasso, Dr. Mohammadreza Amel Solouki, Dr. Thomas George Valentin Avare, Dr. Leonardo Baldo, Dr. Matteo Bertone, Dr. Fabio Bima, Dr. Francesco Cartelli, Dr. Simone Cavariani, Dr. Yushuo Chang, Dr. Pietro Chiavassa, Dr. Mattia Conticello, Prof. Matteo Davide Lorenzo Dalla Vedova, Dr. Francesco De Pace, Dr. Nicola Dilillo, Dr. Matteo Di Salvo, Dr. Luigi Di Sergio, Dr. Federico Fedi, Dr. Matteo Ferrenti, Prof. Renato Ferrero, Dr. Mohammad Ghazi Vakili, Dr. Edoardo Giusto, Dr. Moreno La Quatra, Prof. Paolo Maggiore, Prof. Paolo Montuschi, Dr. Maurizio Negro, Dr. Dario Oppici, Dr. Pier Paolo Politi, Dr. Fabio Porru, Dr. Luigi Pugliese, Dr. Gaetano Quattrocchi, Dr. Gustavo Adolfo Ramirez Espinosa, Dr. Federico Raso, Dr. Luigi Ruggieri, Dr. Antonio Scaldaferri, Dr. Sergio Schiavello, Dr. Jacopo Sini, Prof. Matteo Sonza Reorda, Dr. Simone Tavella, Prof. Massimo Violante, Dr. Luca Viscanti, and Dr. Mengdi Yang.

Abstract

The proliferation of sensors and Internet of Things (IoT) devices has led to the generation of vast amounts of data. Extracting valuable information from these complex data structures often relies on advanced visualization techniques. When appropriately designed, effective data visualization empowers researchers and analysts to identify and interpret key features of the variables involved intuitively. Examples can be the presence of intricate relationships, repeating patterns, and more. However, in scenarios where the information of interest is not readily visible or discernible, multivariate analysis techniques can be employed to extract relevant insights from the data before creating the visualization itself.

Such techniques, among which it is possible to mention factor, cluster, and discriminant analysis, can help uncover hidden patterns, correlations, and anomalies within large datasets, thereby enabling a more comprehensive understanding of the underlying phenomena. Researchers can create sophisticated and interactive visual representations that effectively communicate complex insights and facilitate knowledge discovery by integrating multivariate analysis techniques with data visualization tools. Moreover, if the information of interest remains elusive or requires a more automated approach, Machine Learning (ML) techniques can be further employed to discover hidden patterns and relationships within the data. ML-based approaches, such as Artificial Neural Networks (ANN), Decision Trees (DT), and Support Vector Machines (SVM), can be used to identify complex, nonlinear relationships within data. ANNs, in particular, are designed to simulate the way the human brain processes information, allowing them to learn and improve automatically from experience without being explicitly programmed. Unlike traditional Artificial Intelligence (AI) techniques, ANNs have been able not only to match but also to outperform humans in specific task-oriented applications.

Multivariate analysis, ML techniques, and data visualization can be applied across various domains, including healthcare, finance, education, and transportation, to

address a wide range of challenges and uncover new opportunities for innovation and growth. In this thesis, such technologies were used to analyze various topics of interest for research and industry. Data from Fiber Bragg Grating (FBG) sensors, image sensors, Photoplethysmography (PPG) sensors, and Supervisory Control And Data Acquisition (SCADA) systems were analyzed in multiple ways to get valuable information. Collaborations between departments of the Politecnico di Torino, such as that between the Department of Control and Computer Engineering (DAUIN) and the Department of Mechanical and Aerospace Engineering (DIMEAS), with student teams of the Politecnico di Torino, such as ICARUS, and with companies, such as Sirius and ReLearn, were crucial to the development of the thesis. The discussion on the different topics was divided according to the type of sensor mainly used. Multiple scientific contributions were made to each of them, which will be explained in detail in the thesis itself.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 A World in Continuous Evolution

The geological epoch in which we live, which with some objection is also sometimes called *Anthropocene* [1], is characterized by rapid changes in daily life. As a personal example, my parents were born in the 1950s: the computers of that period were huge, slow, and expensive, so only a few countries or large companies could afford them. The same can be said for telephones, which, in addition to being larger than today's, were operated only by twisted-pair cable. The characteristics of modern computers can hardly be compared to those of that period, as they have become incredibly more powerful, smaller, and cheaper. The same can be said for telephones, which have been replaced almost entirely by smartphones, and multiple other electronic and mechanical devices. During the average lifetime of a human being, it is therefore possible to witness numerous technological changes that can have severe repercussions on lives themselves.

Although with some criticisms [2, 3], it has been stated that this rapid sequence of technological changes in recent history, often defined as *accelerating change*, seems not linear, but exponential [4]. At the current rate, the progress that will occur in the 21st century could be equivalent to that of 20,000 years [5]. This observation was later put into writing in the *Law of Accelerating Returns* by Ray Kurzweil [6]. The problem with these sudden technological changes is that they may or may not be accompanied by an equally sudden cultural and social change [7, 8].

One school of thought in psychology, known as *evolutionary psychology*, focuses on

how the human mind has evolved to enable survival and procreation. It is founded on several assumptions, one of which implies that the brain has developed specific neural systems meant to address issues repeated throughout long evolutionary time, giving modern humans Stone Age minds [9, 10]. Consequently, since our society has made enormous progress in a short time, our brains cannot fully follow these changes and continue to behave as if we were in prehistoric times. Moreover, the report *Global Change and the Earth System: A Planet Under Pressure*, released in 2004 by the International Geosphere-Biosphere Programme (IGBP), has evidenced that, in the mid-20th century, a decisive anthropogenic environmental change occurred throughout the world, referred to as *The Great Acceleration* [11]. Data collected by the IGBP and contained in the report show the evolution of these trends from 1750 to 2010. Figure 1.1 provides a visual explanation of this phenomenon, grouping socio-economic and Earth system trends. For most indicators, it is possible to see a marked acceleration from the second half of the 20th century.

Our world is constantly changing, and often not for the better: rapid technological evolution should be accompanied by an equally profound social, cultural, and economic transformation that respects our planet.

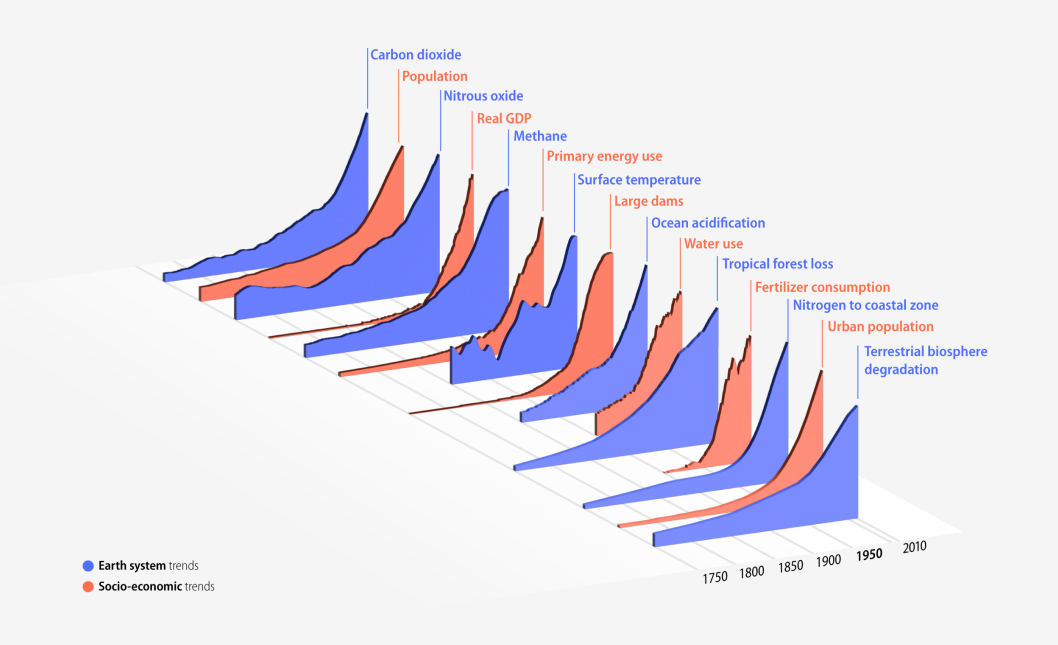

FIGURE 1.1: Since the second half of the 20th century, the Great Acceleration has radically transformed planet Earth and its climate system. This image was taken from [12] and is based on data from [13].

1.2 Data

Data have enabled rapid progress in many technological fields and, together with many other factors, including the steady increase in the computational capabilities of our computers, have made possible the progression of Artificial Intelligence (AI) at current levels. Since the beginning of the 21st century, their importance has increased more and more. In 2006, the mathematician Clive Humby defined them as *the new oil*. Oil itself has little value, but it gains value when it is transformed into plastic, fuel, and more. Likewise, raw data have little value, but if analyzed and processed, their value increases more and more. Due to this reason, data are now widely collected and used in different contexts [14, 15].
Large companies like Meta, Google, Amazon, and Netflix have historically used data for business purposes. For example, data on user behavior can be used to propose targeted advertisements or suggest new movies or series to watch. However, the data can be used meaningfully in many other cases; to name a few, they allow to:

- *Monitor the health status of people and any system*: data are a source of knowledge. Suppose normal and abnormal behaviors can be characterized. In that case, it is possible to intercept any behavior that deviates from normal before it can aggravate the condition itself.
- *Make more informed decisions*: having valuable data can be used as indisputable evidence or justification for making a confident decision. Unlike hypotheses, abstract observations, and anecdotal evidence, data management is more complex and allows for less risky decisions based on prior knowledge.
- *Solve problems*: the causes of failures of systems, business processes, or others can be better understood when examining the data attached to the processes involved. This understanding is the basis for constructing more effective solutions to avoid or reduce the probability that these problems will recur again.
- *Improve a process*: data can allow a better understanding of a certain process. By highlighting obstacles, interruptions, or weaknesses, it is possible to improve it and reduce the number of wasted resources.

There are numerous other reasons, which will not be explored here, why data are considered essential. Instead, the intended message is that, in our society, data have now become vitally crucial in virtually all areas of work and otherwise.

1.3 Sensors

In the previous Section, some examples were cited regarding the usefulness of data in our society. However, such data must often be acquired through *sensors*, which today are virtually everywhere and perform a wide variety of tasks.
Examples of their use include:

- Turning lights on and off at dusk and dawn.
- Detecting smoke for fire prevention purposes.
- Adjusting the temperature of a room according to our preference.
- Automatically braking vehicles when obstacles are detected.

FIGURE 1.2: Some of the sensors currently available on the market. This image was taken from [16].

Many types of sensors are currently available on the market, some of which are illustrated in Figure 1.2.
In practical terms, a sensor is a device that, upon detecting a physical phenomenon of interest (often defined as stimulus or measurand, e.g., light, sound, heat), responds by releasing a corresponding electrical signal [17–19]. In contrast, the device that performs the opposite task of converting a signal, usually electrical, into an action, usually mechanical, is called *actuator* [20]. Both sensors and actuators are a specific case of the more general term *transducers*, which groups together all devices capable of converting whatever kind of energy into another. The loudspeaker is an example of a transducer, as it translates an electrical signal into an alterable magnetic field, after which it produces acoustic waves. However, when connected to the input of an amplifier, it can operate as an acoustic sensor, i.e., a microphone [18].
Figure 1.3 illustrates some of the most commonly utilized sensors in modern vehicles. Continued pressures to improve travel safety in cars and other vehicles have led to the continual addition of sensors and other electronic devices designed to regulate their operation.

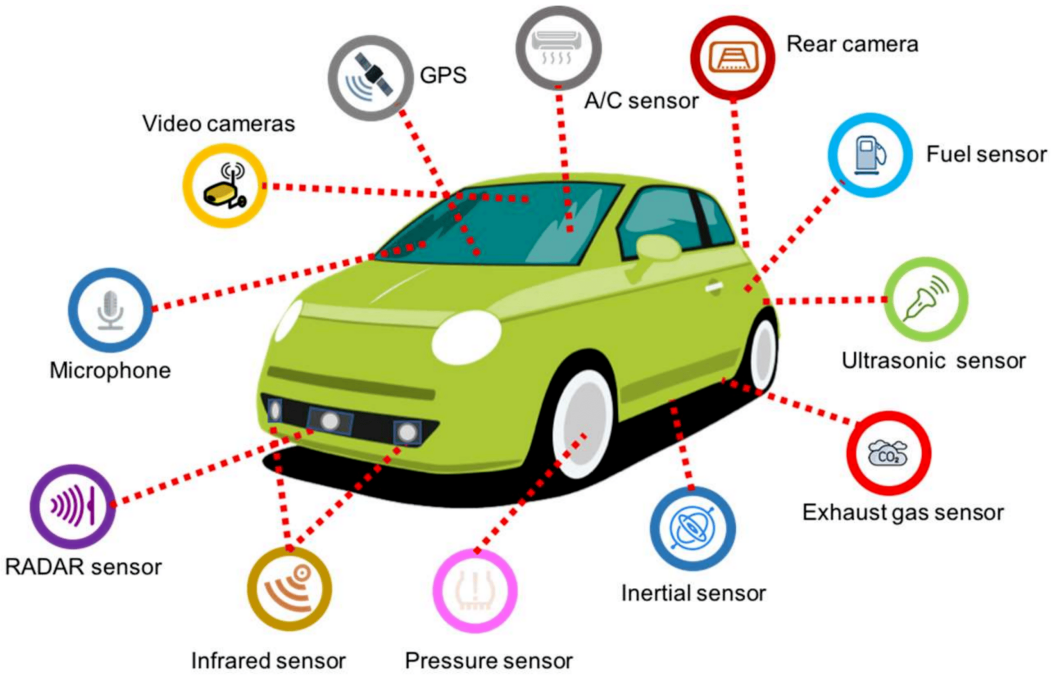

FIGURE 1.3: A visual representation of the most common on-board sensors in contemporary vehicles. This image was taken from [21].

1.3.1 Sensor Classification

Sensor classification schemes can range from very simple to very complex [18]. The choice of classification criteria depends on the specific purpose of the classification. Different purposes may require different criteria to classify and understand different types of sensors effectively.

1.3.2 Simple and Complex Sensors

Sensors can firstly be categorized as *simple* (or *direct*) and *complex* [18].
Simple sensors convert a stimulus directly into an electrical signal or, alternatively, use a suitable physical effect to modify an electrical signal. Examples include photodiodes, thermocouples, and piezoelectric sensors.
On the other hand, complex sensors require one or more energy transducers before a direct sensor can be employed to generate an electrical output. In other words, complex sensors involve multiple stages of energy conversion or signal processing before obtaining the desired electrical signal. Examples include Complementary Metal-Oxide Semiconductor (CMOS) and Charge-Coupled Devices (CCD) image sensors, which involve multiple layers and components to convert incoming light into an electrical signal. Figure 1.4 depicts a complex sensor.

1.3.3 Passive and Active Sensors

Sensors can also be classified as *passive* and *active* sensors [18].
Passive sensors do not require an additional energy source and directly generate an electrical signal in response to an external stimulus. This is because the sensor

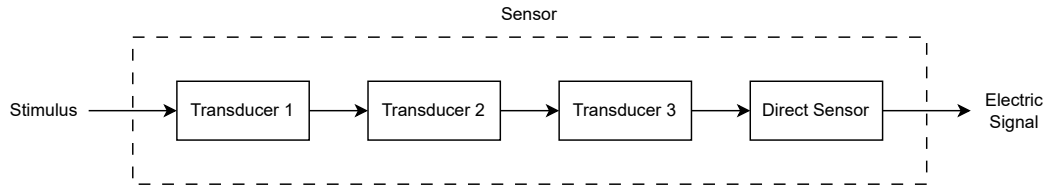

FIGURE 1.4: A complex sensor may consist of multiple transducers, each converting one form of energy into another. The final stage typically involves a direct sensor, which produces an electrical output.

converts the input stimulus energy into the output signal. Most passive sensors are direct sensors; examples of passive sensors include photodiodes, thermocouples, and piezoelectric sensors.
Instead, active sensors need external power for their operation, often mentioned as an excitation signal. The sensor modifies this signal to produce the output signal. Active sensors are sometimes referred to as *parametric* sensors because their own properties change in response to an external effect, and these properties can be subsequently converted into electrical signals. In other words, a sensor parameter modulates the excitation signal, which carries information about the measured value. An example of an active sensor is a resistive strain gauge, where electrical resistance corresponds to strain. An electric current must be applied to a sensor from an external power source to measure its resistance.

1.3.4 Absolute and Relative Sensors

Another possible classification is between *absolute* and *relative* sensors [18].
Absolute sensors detect a stimulus in relation to an absolute physical scale, which is independent of the measurement conditions. In contrast, relative sensors produce a signal that is related to a specific reference point or a particular case. An example of absolute sensor is the thermistor, which is a temperature-sensitive resistor. Its electrical resistance directly corresponds to the absolute temperature scale of Kelvin. On the other hand, the thermocouple, another popular temperature sensor, is a relative sensor. It generates an electric voltage that is a function of the temperature gradient across the thermocouple wires. Therefore, its output signal cannot be associated with a particular temperature without referring to a known baseline. Another instance of absolute and relative sensors is the pressure sensor. An absolute pressure sensor generates a signal in reference to a vacuum, which is the absolute zero on a pressure scale. Meanwhile, a relative pressure sensor produces a signal concerning a selected baseline that is not zero pressure, such as atmospheric pressure.

1.3.5 Property-Based Classifications

Another approach to examining a sensor is to focus on specific properties that may be of particular interest [18]. Listed below are some of the characteristics of sensors that can be considered:

- *Stimulus*: acoustic (e.g., phase, wave amplitude, polarization), electric (e.g., current, voltage, conductivity), mechanical (e.g., position, acceleration, mass), and more.
- *Sensor specifications*: sensitivity (e.g., stability, accuracy, speed of response) and stimulus range (e.g., resolution, selectivity, linearity).
- *Sensor material*: inorganic (e.g., conductor, semiconductor, biological substance) and organic (e.g., insulator, liquid gas or plasma)
- *Conversion phenomena*: physical (e.g., thermoelectric, photoelastic, thermomagnetic), chemical (e.g., chemical or physical transformation, spectroscopy), biological (e.g., biological or physical transformation, spectroscopy).
- *Detection means used in sensors*: biological, chemical, electric, and more.
- *Field of applications*: energy, medicine, transportation, and more.

1.4 Multivariate Analysis

Multivariate analysis encompasses all statistical approaches that inspect several measurements on individuals or objects under examination simultaneously [22]. Therefore, the contemporaneous analysis of two or more variables can be broadly considered a multivariate analysis. Those techniques are widely used and allow organizations to obtain valuable insights and improve their decision-making processes. Many multivariate techniques are generalizations of techniques used in univariate and bivariate analyses. For example, the single dependent variable found in the analysis of variance is extended to include multiple dependent variables in the multivariate analysis of variance. The same can be said for simple regression (with one predictor variable), which is extended to include several predictor variables in the multivariate case. Similarly, some multivariate techniques, such as multivariate analysis of variance and multiple regression, allow for performing a single analysis, which previously required multiple univariate analyses. However, other multivariate techniques are specifically designed to address multivariate issues. For instance, discriminant analysis differentiates among groups based on a set of variables, while factor analysis identifies the underlying structure of a set of variables. These techniques provide valuable insights that cannot be obtained through univariate or bivariate analyses

alone.
The extension to the multivariate domain introduces additional concepts and issues that are particularly relevant. These concepts range from understanding the basic building block of multivariate analysis, the variate, to specific issues related to measurement scales and statistical significance testing and confidence levels. Each concept plays a crucial role in successfully applying any multivariate technique.

1.4.1 Variate

The *variate*, a linear combination of variables with empirically determined weights, is the building block of multivariate analysis [22]. The researcher specifies the variables, while the multivariate technique determines the weights to achieve a specific objective. A variate of n weighted variables (X_1 to X_n) can be represented mathematically as:

$$Variate\ value = w_1X_1 + w_2X_2 + w_3X_3 + ... + w_nX_n \tag{1.1}$$

where w_n is the weight determined by the multivariate technique and X_n is the observed variable. The result is a single value combining the entire set of variables that best meets the aim of the specific multivariate analysis.
In multiple regression, the variate is determined in a way that maximizes the correlation between the single dependent variable and the multiple independent variables. In factor analysis, the variates are formed to represent best the underground structure or patterns of the variables, represented by their interrelationships. On the other hand, in discriminant analysis, the variate is formed to make scores for each observation that differentiate the groups of observations more closely. In any case, the variate captures the multivariate nature of the analysis, so it is its focal point in many respects. It is essential to understand its collective impact in achieving the goal of the technique and any contribution of the separate variable to the overall effect of the variate.

1.4.2 Measurement

Data analysis involves identifying and measuring variation in a set of variables, either between a dependent variable and one or more independent variables or between

them [22]. The critical aspect of this process is *measurement* since the researcher can only identify variation by measuring it. Measurement is essential to accurately represent the concept of interest and is critical for selecting the most appropriate multivariate analysis method. Data can be classified differently based on their characteristics and attributes:

- *Metric* (or *quantitative*) data are used when subjects differ in degree or quantity on a particular attribute. Metrically measured variables reflect a relative degree or quantity, making them suitable for attributes involving magnitude or quantity, such as commitment to a job or level of satisfaction.
- *Nonmetric* (or *qualitative*) data describe differences in type or category by indicating the presence or absence of a property or characteristic. These properties are discrete, which means that if an object or individual has a particular characteristic, all other characteristics in that category are excluded. For example, if a car and a motorcycle are considered, they are both vehicles, but the car cannot be simultaneously a motorcycle and vice versa. There is no quantity related to the vehicle, only the implied status given by its kind.

The researcher must define the type of measurement of each variable as metric or nonmetric. To the computer, these values are just numbers; however, describing the data as metric or nonmetric significantly impacts what the data can represent and how it can be analyzed.

1.4.3 Extract Useful Information from Data

Large amounts of data, such as those produced by sensors, can be used to perform multivariate analysis and reveal the presence of hidden links between the same data. For such analysis, visualization is often the most essential and comprehensive tool because, if appropriately designed, it allows some of the features of the variables involved to be noted intuitively. Where the information of interest is not readily visible, multivariate analysis and Machine Learning (ML) techniques can be used to extract it from the data before creating the visualization itself. Some of the commonly used multivariate analysis techniques are, for example, the following [23]:

- *Multiple linear regression*: is used to analyze the relationship between one dependent variable and two or more independent variables. For example, if a

couple trying to sell their house is considered, the dependent variable is the price of the house. In contrast, the independent variables are location, time of year, interest rates, and the existence of similar listings in the area. In the end, the price of the house will depend on the combination of these factors. Multiple linear regression can be used to determine the strength and direction of the relationships between the dependent and independent variables and to develop a predictive model that can be used to estimate the value of the former based on the values of the latter.

- *Multiple logistic regressions*: is used to analyze the relationship between a categorical dependent variable with more than two levels and one or more independent variables. Possible examples are the possibility that the partner accepts a marriage proposal or that a child graduates from college. Many independent variables could influence these dependent variables, such as the partner's previous expressions of interest in marriage, the child's study habits, and more. The objective of multiple logistic regression is to estimate the probability that the dependent variable takes on a continuous or discretized value based on the values of the independent variables. As another example, this technique can be helpful for a contractor trying to predict the likelihood that a particular type of customer will contract with his demolition company for a job.

- *Multivariate Analysis of Variance (MANOVA)*: is used to analyze the difference in the effect of multiple independent variables on multiple dependent variables. As an example, it proves helpful if a marketer wants to increase sales of a face cream by studying the impact of two independent variables (price reduction and campaign budget increase) on two dependent variables (face cream sales and total revenue). MANOVA allows the effect of each independent variable on each dependent variable to be tested simultaneously while controlling for the effects of the other independent variables. It thus fosters an understanding of the complex relationships among multiple variables and how they interact with each other to influence the outcome.

- *Factor analysis*: is used to reduce the number of variables in a dataset by identifying and grouping together closely correlated variables. This can help to simplify the data and make it easier to identify patterns and relationships. Factor analysis could, for example, be used to combine income and education

into a single variable, *socioeconomic status*, or multiple behaviors, such as leaving a positive review and signing up for a newsletter, into a single variable, *customer loyalty*.

- *Cluster analysis*: is used to group similar observations or data points based on their characteristics or traits. It is often used in exploratory data analysis to identify patterns or structures. In market segmentation, cluster analysis can, for example, be used to group customers or potential customers into segments based on their similarities in relevant behaviors, such as purchase history, browsing behavior, or demographic information. This way, marketers can tailor messages and offers to each segment, delivering a more personalized and relevant experience. Regularly performing cluster analysis and updating customer segments can help teams stay abreast of evolving customer behavior and preferences and ensure the delivery of more effective and relevant marketing campaigns.

- *Discriminant analysis*: is used to classify observations or data points into predefined groups based on their features or traits. Unlike factor and cluster analysis, it is a dependent technique, meaning it starts with a specific dependent variable and examines the impact of the independent variables on the similarity or distinctness between groups. For example, to distinguish between repeat shoppers and non-regular shoppers, the former might represent a dependent variable and discriminant analysis can be used to examine the impact of independent variables such as age, location, or gender. This can help marketers better understand the characteristics of their repeat customers and tailor their marketing efforts to attract and retain more customers.

- *Conjoint analysis*: is a technique used to determine the relative importance of different product or service attributes to consumers. It is widely used in market research and involves surveying consumers and asking them to make trade-offs between different combinations of attributes. It is essential to survey a large and representative sample of consumers from the target audience to ensure that the results of a conjoint analysis survey are accurate and actionable. The survey format should also be carefully designed to avoid overwhelming respondents with too many options while still capturing key nuances in their preferences. One approach to conjoint analysis is *choice-based* conjoint analysis, in which respondents are presented with a series

of full-profile concepts (e.g., winter jackets with different combinations of features) and asked to choose their preferred option. This approach can be more realistic and engaging for respondents but may require multiple rounds of iteration to refine the results.

1.5 Research Objective

The title of this doctoral thesis contains a multiplicity of meanings; at a first attempt, it may take time to imagine the final aim of the research. The final objective of the research can then be summarized as follows:

- Perform a careful analysis of data from different typologies of sensors.
- If applicable, create views that allow users to display them intuitively to extract valuable and unnoticeable information.
- Alternatively, use techniques derived from multivariate analysis and ML to extract these meanings and expose them to the user.
- These operations should be carried out in pure research environments and others with a distinctly industrial impact.

The activities in the field of pure research involved collaboration with various research groups belonging to my department, the Department of Control and Computer Engineering (DAUIN), but also with other departments of the Politecnico di Torino, in particular with the Department of Mechanical and Aerospace Engineering (DIMEAS). Those having an industrial impact have seen the collaboration with two companies: Sirius [24], which develops tools for energy management, and ReLearn [25], which creates intelligent solutions for waste monitoring.
These activities have led to the creation of a multitude of contributions, which will be minutely described in the different chapters of this thesis and the conclusion.

1.6 Thesis Outline

Since different typologies of sensors have been used in very different fields, it was decided to divide the discussion according to the sensors employed. The outline of the thesis is, therefore, as follows:

- Chapter 2 reports the contributions relative to the use of Fiber Bragg Grating (FBG) sensors.
- Chapter 3 shows the contributions concerning the use of image sensors.
- Chapter 4 shows the contributions concerning the use of Photoplethysmography (PPG) sensors.
- Chapter 5 presents the contributions pertaining to the use of data from Supervisory Control And Data Acquisition (SCADA) systems.
- Chapter 6 reports conclusions about the entire doctoral work and some hints about possible improvements to be made in the future.

Chapter 2

Fiber Bragg Grating Sensors

This chapter discusses a project carried out in collaboration with the Department of Mechanical and Aerospace Engineering (DIMEAS) department and the ICARUS student team of the Politecnico di Torino [26], which involved the use of Fiber Bragg Grating (FBG) sensors for the monitoring of an Unmanned Aerial Vehicle (UAV).

2.1 Optical Fiber

FBG sensors are sensors built into an *optical fiber*. This Section will, therefore, give details regarding the operation, advantages, and applications of optical fibers and these types of sensors.

2.1.1 Characteristics and Operation

A standard optical fiber typically includes a cylindrical glass core surrounded by a glass cladding, having a total diameter of $125\mu m$, with one or more additional polymer buffer coatings providing increased protection from mechanical and environmental factors [27, 28]. Considering the outer coating, which has a polymer composition that serves a structural function, the total diameter of the fiber reaches $250\mu m$. Optical fiber is used to transmit light rays over short and long distances; this is possible thanks to the particular composition of the core and cladding. After entering the fiber, the light remains confined to its core, which has a diameter of approximately $8\mu m$ [29].

It is crucial to introduce some notions to understand how light is transmitted inside the optical fiber. When a light ray encounters a boundary between two materials with different refractive indices (e.g., glass and air), part of the ray is reflected in the first medium. In contrast, the remaining part is bent (or refracted) when it enters the second material. This phenomenon is governed by *Snell's law of refraction*, which relates the angles of incidence and refraction at the interface between the two materials. It states that:

$$n_1 sin\phi_1 = n_2 sin\phi_2 \tag{2.1}$$

According to the law of reflection, the angle ϕ_1 at which the incident ray strikes the interface is equal to the angle ϕ_3 at which the reflection ray is produced, and both rays lie in the same plane perpendicular to the interface. This plane is known as the *plane of incidence*.

The reflection of light off a material that is optically denser (with a higher refractive index) than the material it travels through is called external reflection. In contrast, the reflection of an optically less dense material (for example, light in glass reflected at a glass-air interface) is called *internal reflection*.

As the angle of incidence ϕ_1 in an optically denser material increases, the refracted angle ϕ_2 will gradually tend toward $\frac{\pi}{2}$ (90°). Beyond this *critical angle of incidence* ϕ_c no refraction occurs and total internal reflection occurs, allowing the light to

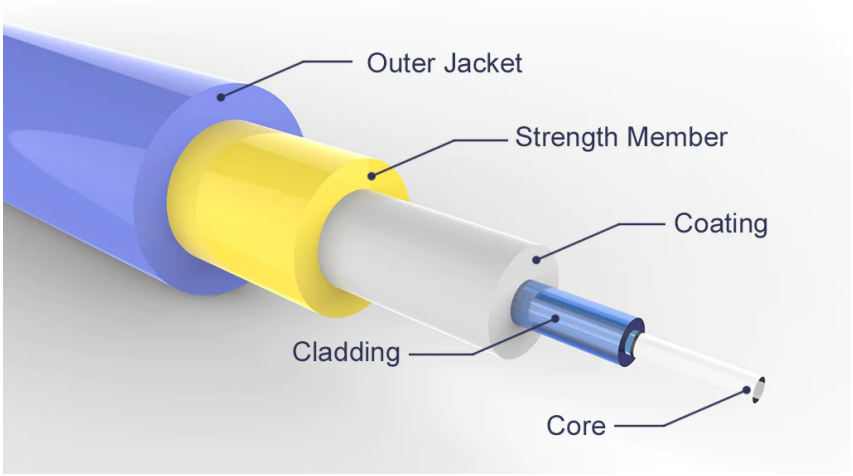

FIGURE 2.1: The optical fiber waveguide consists of a core with a refractive index n_1, encased by a cladding with a slightly lower refractive index n_2. Additional coatings can be added to provide additional mechanical protection to the fiber and help preserve the waveguide. This image was taken from [30].

remain entirely in the fiber. This angle can be determined using Equation (2.1):

$$\frac{sin\phi_1}{sin\phi_2} = \frac{n_2}{n_1} \tag{2.2}$$

which can be rewritten as follows:

$$sin\phi_c = \frac{n_2}{n_1} \tag{2.3}$$

When the *angle of incidence* exceeds the critical angle of incident, all the light is reflected into the denser material without any light escaping into the adjacent medium. This is exactly the desired working situation for optical fibers, in which the incoming light must not be scattered toward the outside of the fiber.
Based on what has been described above, it is easy to understand that only a finite set of rays with discrete angles greater than or equal to the critical angle ϕ_c can propagate along the fiber. These angles are related to different field distributions or patterns of electromagnetic waves called *modes*, relating to how the rays can travel along the fiber. A fiber is defined as *single-mode* if only the fundamental ray can propagate

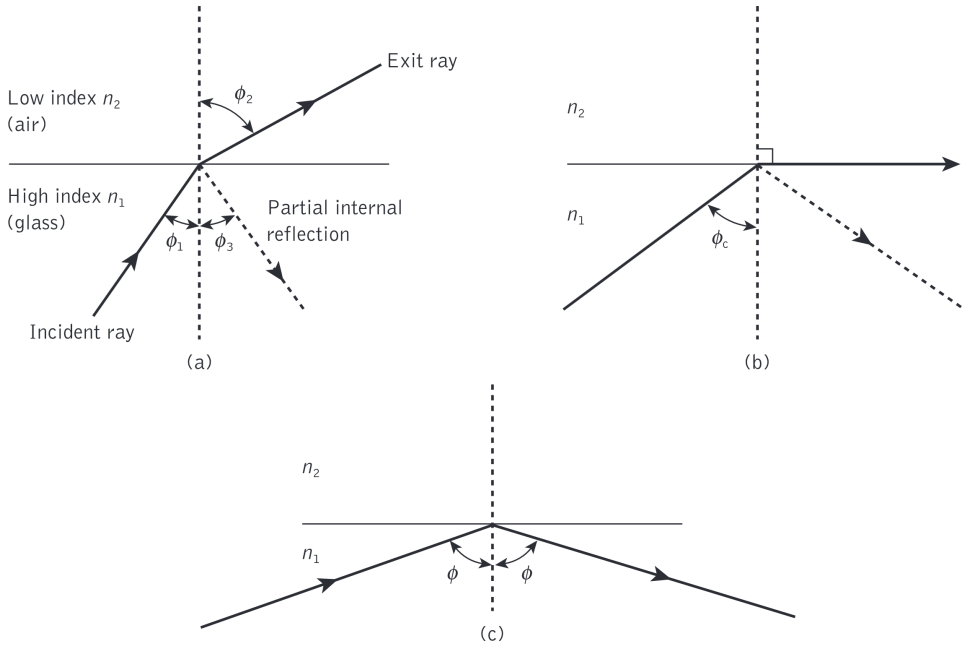

FIGURE 2.2: Light rays incident on an interface (e.g., glass-air) with a high to low refractive index: Refraction (a), the limiting case of refraction displaying the critical ray at an angle ϕ_c (b), and the case of total internal reflection where $\phi > \phi_c$ (c); This image was taken from [28].

directly along the fiber axis. This feature can be achieved if the core diameter is small, with values around $8-10\mu m$. On the other hand, *multimode* fibers support many beams or propagation modes due to larger core diameters, usually greater than or equal to $50\mu m$. Multimode fibers can be further divided as follows:

- *Step-index* fibers: they have a constant refractive index along the entire section of the core and change suddenly when it encounters the cladding.
- *Graded-index* fibers: they have a refractive index that gradually changes from core to cladding, allowing multichromatic light.

Single-mode fibers are much cheaper than multimode fibers, can cover longer distances, and reach significantly higher speeds. On the other hand, they require more expensive laser-emitting devices and more precise and accurate terminal wiring. Their ability to support only one propagation mode makes them ideal for applications where minimizing signal dispersion and maximizing bandwidth are critical, such as long-range telecommunications networks and high-speed data transmission systems. On the other hand, multimode fibers allow the use of cheaper devices, such as Light-Emitting Diode (LED), Edge Emitting Light-Emitting Diode (ELED), and Vertical Cavity Surface Emitting Laser (VCSEL), but suffer from the phenomenon of intermodal dispersion, whereby the different modes propagate at slightly different

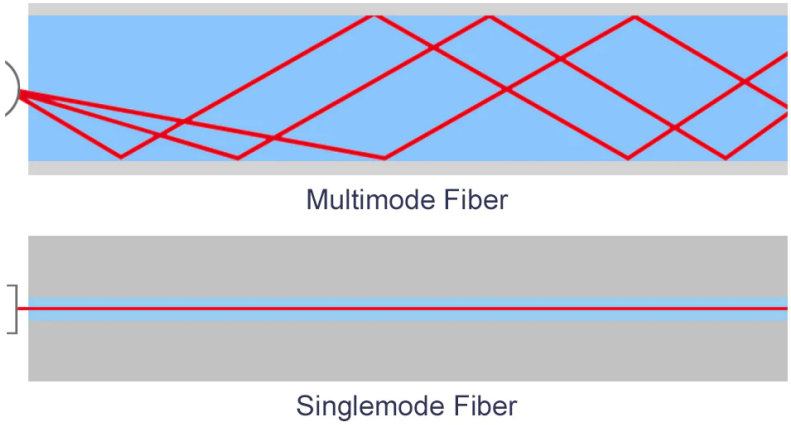

FIGURE 2.3: In single-mode fiber optic cables, only one light mode can be propagated at a time, while multimode fibers can propagate multiple modes simultaneously. Key differences include wavelength, light source, fiber core diameter, bandwidth, transmission distance, cost, and colored sheath. This image was taken from [30].

times, and this limits the maximum distance at which the signal can be received correctly, especially at high speed (Gb/sec). Multimode fibers are commonly used in Local Area Network (LAN) applications within companies or government agencies. They are particularly suitable for Gigabit Ethernet (GigE) and 10 Gigabit Ethernet (10GigE) links, where the distances involved are shorter, and the emphasis is on high-speed data transfer within a confined area.

2.1.2 Advantages

Optical fiber has numerous advantages, some of which are listed below [27, 28]:

1. *Small size and weight*: optical fibers possess minimal diameters, often comparable to the diameter of a human hair. Optical fibers remain significantly smaller and lighter than their copper counterparts, even when coated with protective layers. This feature offers substantial benefits, particularly in urban environments where duct congestion can be a significant problem. Additionally, the compact nature of fiber optics allows for extended signal transmission capabilities in mobile applications such as ships, aircraft, and satellites.

2. *Interference immunity*: optical fibers act as dielectric waveguides, making them immune to Electromagnetic Interference (EMI), Radio-Frequency Interference (RFI), or switching transients that produce Electromagnetic Pulses (EMPs). As a result, the operation of a fiber optic communications system is not affected by transmission through electrically noisy environments, so the fiber cable does not require shielding from EMI. Unlike communication using electrical conductors, it is also relatively simple to prevent optical interference between fibers, ensuring minimal crosstalk even when multiple fibers are wired together. In addition, if fiber cables are used overhead rather than underground, the fiber cables are not susceptible to lightning strikes.

3. *Enormous potential bandwidth*: optical fibers offer immense transmission bandwidth potential due to their optical carrier frequency, typically in the range of $10^{13}Hz$ to $10^{16}Hz$ and commonly in Near Infrared (NIR) spectrum around $10^{14}Hz$. This bandwidth far exceeds that of metallic cable systems, such as coaxial cables with typical bandwidths of around $20MHz$ over distances up to $10km$, or even millimeter wave radio systems operating with bandwidths

of $700MHz$ modulation over shorter distances. By 2000, fiber optic links incorporating fiber amplifiers demonstrated a bandwidth length product of $5000GHz\,km$, a notable improvement over coaxial cable systems with a typical bandwidth length product of approximately $100MHz\,km$. This results in a 50000 bandwidth improvement over the coaxial cable while providing superior information-carrying capacity over much longer transmission distances.

4. *Electrical isolation*: being made of glass or a plastic polymer, optical fibers act as electrical insulators. This property eliminates the ground loop and interface problems commonly associated with metallic counterparts. Because the optical fibers pose no risk of arcing or sparking in the event of abrasion or short-circuiting, they are also ideal for communication in electrically hazardous environments.

5. *Signal security*: Optical fibers offer a high level of signal security due to the minimal light radiation. Unlike copper cables, it is difficult to intercept an optical signal from a fiber without drawing optical power, making noninvasive interception virtually impossible. This inherent feature provides increased security for military, banking, and general data transmission applications, as any attempt to intercept optically transmitted messages will likely be detected.

6. *Low transmission loss*: Over the past two decades, optical fibers have made significant progress, producing cables with extremely low attenuation or transmission loss compared to copper conductors. Some fibers have been built with losses as low as $0.15dB\ km^{-1}$. This characteristic has become one of the significant advantages of fiber optic communications, enabling the implementation of communication links with large distances between repeaters or optical amplifiers and consequent reduction in cost and system complexity. Together with the proven modulation bandwidth capability of fiber cables, this feature has made the adoption of fiber optic communications compelling in most long-haul telecommunications applications. This has led not only to the replacement of copper cables but, to some extent, also to satellite communications because of the significant delay involved in this technology for the transmission of voice and other signals.

7. *Robustness and flexibility*: Despite the need for protective coatings, optical fibers can be produced with very high tensile strengths. Despite being made of

glass, optical fibers can be twisted or bent into small spokes without damage. In addition, cable structures have been developed to be flexible, compact, and extremely strong. In general, fiber optic cables outperform corresponding copper cables in storage, transportation, handling, and installation, as they have size and weight advantages and at least comparable strength and durability.

8. *System reliability and ease of maintenance*: fiber optic cables reduce the need for intermediate repeaters or line amplifiers to increase the transmitted signal strength as they have low loss. Compared with conventional electrical conductor systems, system reliability is typically improved because fewer repeaters or optical amplifiers are needed. Having an expected lifetime of 20-30 years, the reliability of optical components is also relatively high. These factors also reduce the maintenance time and costs associated with fiber optic systems.

9. *Low-cost potential*: Because the primary material for making optical fiber is glass made from sand, which is not a scarce resource, it offer the potential for lower-cost line communications compared to copper conductors. While fiber optic manufacturing costs have become competitive with copper wire for bulk purchases, other components associated with fiber optic communications, such as high-performance semiconductor lasers and detector photodiodes, still have relatively high costs. In addition, some connection technologies, such as removable connectors and couplers, contribute to overall system costs.
   Overall system costs for long-haul links using fiber optic communication are substantially lower than those of equivalent power line systems due to the low-loss and broadband properties of the optical transmission medium. Nonetheless, the additional cost due to electrical-optical conversion can be a deciding factor in short-range applications. However, the potential cost advantages of shipping, handling, installation, and maintenance, along with the inherent benefits of fiber optics, may be significant factors in system selection.

2.1.3 Optical Link Elements

From a simplistic perspective, the function of an optical fiber link is to reliably transport a signal from one electronic device, such as a computer or a telephone, to

corresponding equipment at another location with precision and accuracy [27, 28]. An optical fiber communications link includes:

- *Transmitter*: includes a light source and associated electronic circuitry. The source can be a laser diode or a LED. The electronic components are used to set the operating point of the source, stabilize the light emission, and modulate the optical emission based on an electrical input signal.
- *Optical fiber*: as discussed in Section 2.1.1, is housed inside a cable that provides mechanical and environmental protection. Various types of fiber optics are available, and different configurations are adopted depending on the installation requirements, which may involve indoor environments as well as underground ducts, outdoor poles, or underwater.
- *Receiver*: contains a photodiode that detects the weakened and distorted optical signal coming out from the end of the optical fiber and converts it back into an electrical signal. It also includes amplification devices and circuitry to restore signal integrity.
- *Passive devices*: are optical components that do not require electronic control for their operation. These include optical connectors for cable connections, splices for joining bare fibers, optical isolators to prevent reverse light flow, optical filters for selecting specific light spectra, and couplers for extracting a portion of light, often for monitoring purposes.
- *Optical amplifiers*: Optical signals weaken as they travel along a fiber due to power loss. Optical amplifiers boost the power level of these signals entirely in the optical domain, eliminating the need for conversion to electrical signals for amplification and bypassing transmission bottlenecks.
- *Active components*: Lasers and optical amplifiers are examples of active devices requiring electronic control. Other active optical components include tunable optical filters, light signal modulators, optical switches, and variable optical attenuators.

2.1.4 Applications

The advantages mentioned above make the optical sensors particularly suitable for multiple applications, some of which are highlighted below [31]:

- Measuring physical quantities like strain, temperature, pressure, velocity, acceleration, or object shape.
- Monitoring the health of material and structure, measuring vibrations, and detecting deformation, delamination, and cracking.
- Chemical industry applications include sensing organic chemicals, vapors, gases, pH, humidity, and ions.
- Environmental science applications like environmental analysis, industrial production monitoring, and bioprocesses control.
- Biomedical and biometric applications include long-term health assessment, obtaining fingerprint images, blood flow measurement, skin irritant testing, and stomach and duodenum blood perfusion measurement.
- Automotive applications include light sensing and measurement, pollution detection, implementation of rain and light sensors, gas sensors, wheel rotation sensors, vision sensors, and ultrasonic sensors.
- Industrial applications include surface inspection, composition analysis, product and material characterization, delamination and defect detection, and real-time thermal imaging.

2.2 Fiber Bragg Grating

FBG sensors were used during my Ph.D. These microstructures, whose working is depicted in Figure 2.4, are typically created through photo inscription by Ultraviolet (UV) laser beam of a short segment of optical fiber, usually less than $1cm$. This process induces a permanent change in the physical characteristics of the silica that results in periodic changes in the refractive index of the core. Such an interference pattern thus obtained is known as a *Bragg grating* and operates as a narrowband filter

[29]. In more detail, it acts as a selective mirror that reflects a specific frequency called the *Bragg wavelength* when a beam of light passes through the fiber. In practice, this reflected frequency is the output of the sensor and is defined as:

$$\lambda_b = 2n_{eff} * \Lambda \tag{2.4}$$

where:

- λ_b is the Bragg wavelength.
- n_{eff} is the modified refractive index of the core.
- Λ is the grating pitch.

The proportional relationship between the optical output of the FBG and its geometric characteristics allows for the correlation of wavelength variations with changes in physical quantities acting on the sensor, typically mechanical strain, temperature, pressure, humidity, and more. This research has mechanical strain and temperature measurements as its primary interest so that the following discussion will cover them. Considering only these physical quantities, the general equation governing the response of an FBG sensor can be expressed as follows:

$$\frac{\Delta\lambda}{\lambda_B} = K_T \Delta T + K_\varepsilon \Delta_\varepsilon \tag{2.5}$$

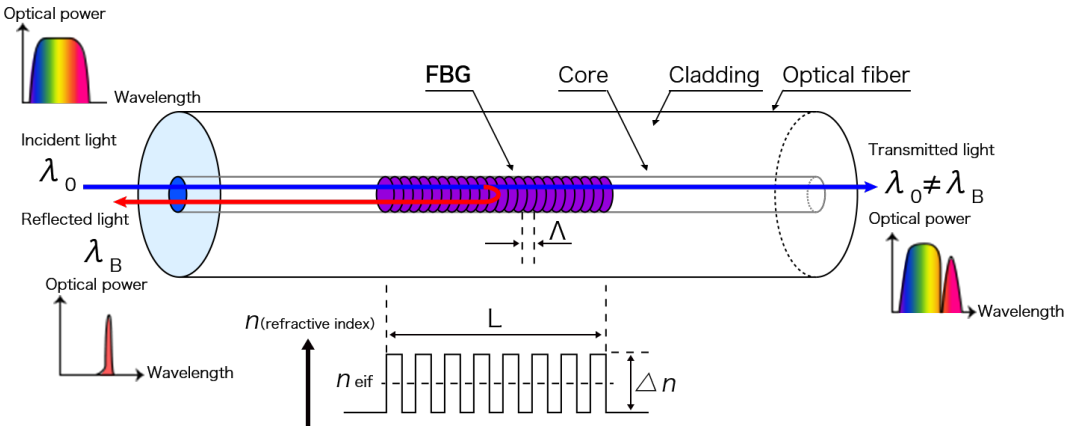

FIGURE 2.4: Structure and working principle of optical fiber and FBG. This image was taken from [32].

where:

- $\Delta\lambda = \lambda_i - \lambda_0$ is the wavelength variation of the measure i from the base value λ_0.
- K_T is the temperature proportionality coefficient.
- $\Delta T = T_i - T_0$ is the thermal variation of the measure i from the base value T_0.
- K_ε is the strain proportionality coefficient.
- $\Delta\varepsilon = \varepsilon_i - \varepsilon_0$ is the strain variation of the measure i from the base value ε_0.

When the optical fiber undergoes traction or compression, the relative period of the microstructure changes, as does the Bragg wavelength. This phenomenon allows FBG sensors to measure the mechanical strain to which the fiber is subjected. For temperature measurements, things are a bit more complicated since removing or compensating for a mechanical strain that might act on the sample is necessary. If this condition is met, the main factor contributing to the change in Bragg wavelength will be the change in the refractive index of the silica induced by the thermo-optical effect. There is also a minor contribution due to thermal expansion changing the microstructure period, which is often negligible due to the low thermal expansion coefficient of silica. In more detail, when sensors are configured to be sensitive only to temperature or strain, they can measure these parameters according to the following equations:

$$T_i = \frac{\Delta\lambda}{(\lambda_B * K_T)} \tag{2.6}$$

$$\varepsilon_i = \frac{\Delta\lambda}{(\lambda_B * K_\varepsilon)} \tag{2.7}$$

2.3 PhotoNext

FBG sensor research is conducted as an internal collaboration at the PhotoNext Interdepartmental Center of the Politecnico di Torino, specializing in photonic technologies [33]. This center, created in 2017 and whose logo is depicted in Figure [34], brings together contributions from five distinct departments:

- Department of Control and Computer Engineering (DAUIN).

- Department of Mechanical and Aerospace Engineering (DIMEAS).
- Department of Electronics and Telecommunications (DET).
- Department of Environment, Land and Infrastructure Engineering (DIATI).
- Department of Applied Science and Technology (DISAT).

Photonics has been acknowledged as a Key Enabling Technology (KET) in the roadmap of the European Commission's Horizon 2020. PhotoNext strategically aligns itself with the European Photonics Work Program, directing its endeavors toward three key areas:

1. Novel optical sensors designed for security, industry, civil services, and life sciences applications.
2. Innovative optical components with the capability to generate, manipulate, and detect light in new ways.
3. Optical networks with ultra-high bit rates to pave the way for the next generation of broadband optical communications.

2.4 Project

The project I worked on during my Ph.D. involving FBG sensors consists of improving a remote monitoring and data visualization system for an UAV equipped with this type of sensor [35–37]. It was developed in collaboration with the DIMEAS department and the ICARUS student team of the Politecnico di Torino [26]. It comprises two primary components:

FIGURE 2.5: PhotoNext Interdepartmental Center logo. This image was taken from [34].

- *Physical system*: includes sensors installed on an aircraft model and the acquisition and telemetry system.
- *Software applications*: handle the reception, transmission, storage, and user-friendly visualization of data coming from the sensors.

Figure 2.6 illustrates a block diagram representing the physical system and software applications, showcasing the connections between diverse hardware components. This project already existed before I started my Ph.D., and the Information Technology (IT) part consisted of the following parts:

- *Middleware* for local data reception.
- *Libraries* for allowing the Middleware to work.
- *Emulator* to facilitate the development of visualization applications.
- *Three Dimensional (3D) Viewer* and an experimental *Augmented Reality (AR) application* for visualizing sensors data.

This was supposed to be the main project of my Ph.D., but the Coronavirus Disease 2019 (COVID-19) pandemic changed the rules of the game, and the work could only begin in earnest in the second year of my Ph.D.

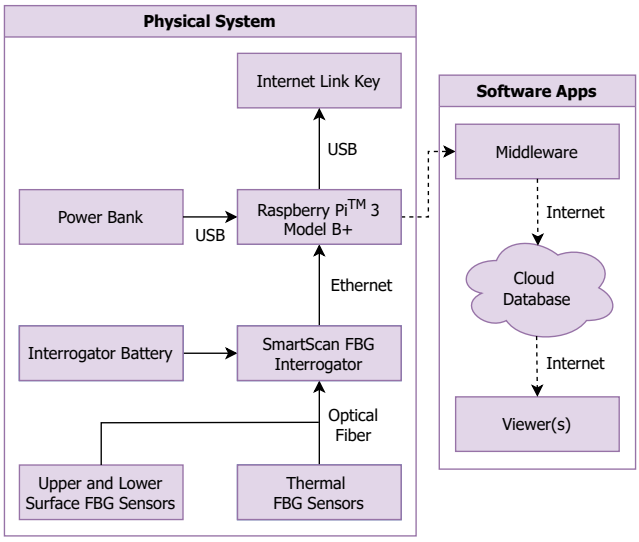

FIGURE 2.6: Illustration depicting the physical system and software applications in a block diagram.

As far as this project is concerned, the main contributions made during my Ph.D. are briefly listed below:

- Middleware can now also send data via the Internet to a *Cloud Database* for storing and sending received data to visualization applications, allowing for a real flying test bench.
- A new application for visualizing data on the desktop has been created, which is lighter and more functional than the previous one.
- A new application for visualizing data via AR has also been created, which is easier to use and has more functions.

The next Sections will contain an extensive description of the parts that make up both the old and the new versions of the project. This will provide a better understanding of how the project works and the additions made during this Ph.D.

2.4.1 Physical System

FBGs were integrated into a UAV called *Anubi*, the first ever built by the ICARUS student team. Anubi was designed to participate in the Air Cargo Challenge (ACC) 2017 competition in Zagreb (Croatia) in 2017. It features a wingspan of 6 meters, electric propulsion, and a Maximum Takeoff Weight (MTOW) of 20 kg. On one half-wing of the UAV, three lines of optical fiber were strategically placed, each serving a distinct purpose:

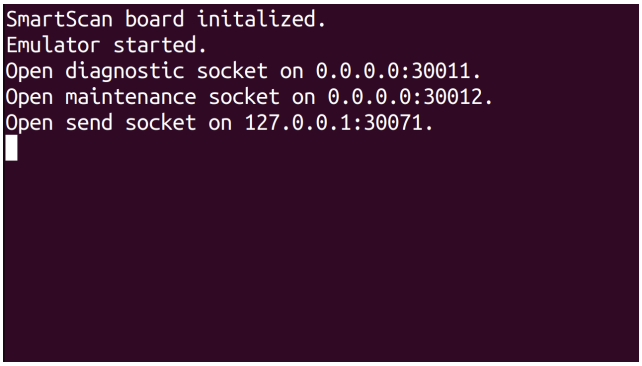

FIGURE 2.7: Upon initiating the program, the Emulator presents a straightforward GUI. This image was taken from [38].

- The first line is located on the upper surface of the wing, which is prone to compression while flying.
- The second line is located on the lower surface of the wing, which is prone to traction while flying.
- The third line, housing only one FBG sensor, allows thermal effects to be taken into account.

Anubi houses the acquisition and telemetry system, comprising the following components:

- An Internet-enabled *Raspberry Pi 3 Model B+* board using an *Alcatel IK40V Fourth Generation (4G) Internet Link Key* with a maximum transmission speed of 150 Mb/s. This board, connected via Ethernet to the interrogator and running Middleware software, receives and forwards data to subsequent system blocks.
- A *Universal Serial Bus (USB) power bank* to supply power to the Raspberry 3 Model B+.
- A *SmartScan FBG interrogator* for reading measurements from FBG sensors installed on Anubi's composite airframe.
- A *9-V Lithium-Ion Polymer (LiPo)* dedicated battery to power the SmartScan FBG interrogator.

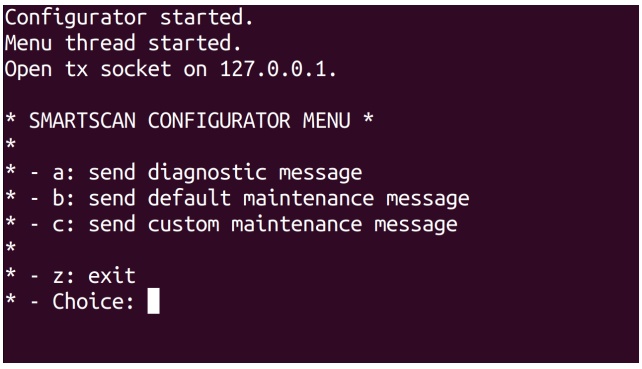

FIGURE 2.8: The Configurator menu allows transmitting various message types utilizing the custom UDP protocol developed by Smart Fibres. This image was taken from [38].

2.4.2 Emulator

Interrogators, being delicate devices, pose challenges in testing due to the complex configurations in which they are often placed. Moreover, their high cost is often a limitation for research teams to purchase them in quantity. The DIMEAS research group with which we collaborate has three different interrogators available: two SmartScan and one SmartScan SBI made by Smart Fibres. As their number is limited, it is only possible to work on the software when they are not used for measurements, which can often last for weeks. To overcome this limitation, an interrogator emulator simply called Emulator was developed to prevent the need for the actual interrogator for the development and testing of software applications [38, 39]. Figure 2.7 illustrates the Graphical User Interface (GUI) of the software. The Emulator behaves precisely as the real interrogator. It transmits to devices connected to its Internet Protocol (IP) and port, not valuable raw and peak wavelength values, but random ones. It sends this type of data and other types of messages, including the current configuration, which helps understand the number of connected sensors and the related channels and gratings. It operates through two abstraction libraries, namely *LibUtils* and *LibSmartScan*, designed to create a unified library extendable to future interrogator models and brands. These libraries internally handle the conversion of user commands, expressed in a language common to all

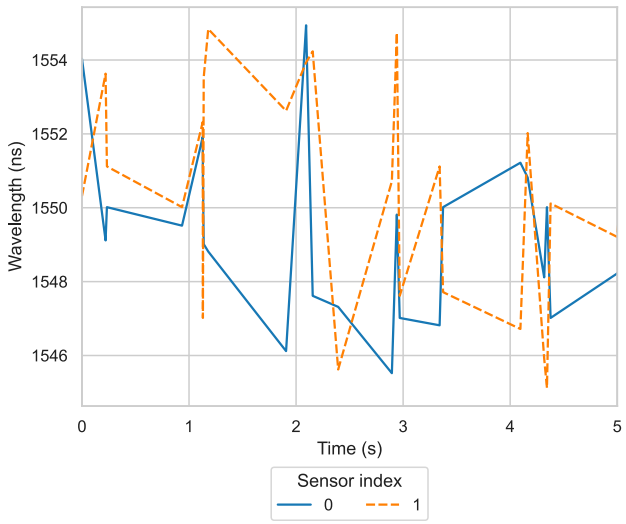

FIGURE 2.9: An illustration of the output generated by the Emulator when tasked with simulating data from 2 sensors linked across one interrogator channel. This image was taken from [38].

interrogators, into commands that can drive a specific emulator. Currently, the only interrogator supported is SmartScan. The creation of the LibSmartScan library was made possible thanks to the datasheets provided by Smart Fibres. It contains details of the customized User Datagram Protocol (UDP) used by SmartScan to send raw and peak wavelength values, modify its behavior, and more. Therefore, any data sent by this Emulator has an identical format to that of this interrogator, with some limitations.

The Emulator offers flexibility in changing various operational parameters, such as the current configuration (number of sensors for each channel), transmission speed, and start and stop of data transmission. These parameters can be configured using another software called *Configurator*, the menu shown in Figure 2.8, which can be run on the same computer or another one on the local network. Figs. 2.9 and 2.10 show examples of output related to data simulation from 2 sensors connected to a single interrogator channel and 16 sensors connected to 4 interrogator channels, respectively.

Both the Emulator and Configurator, together with the LibUtils and LibSmartScan

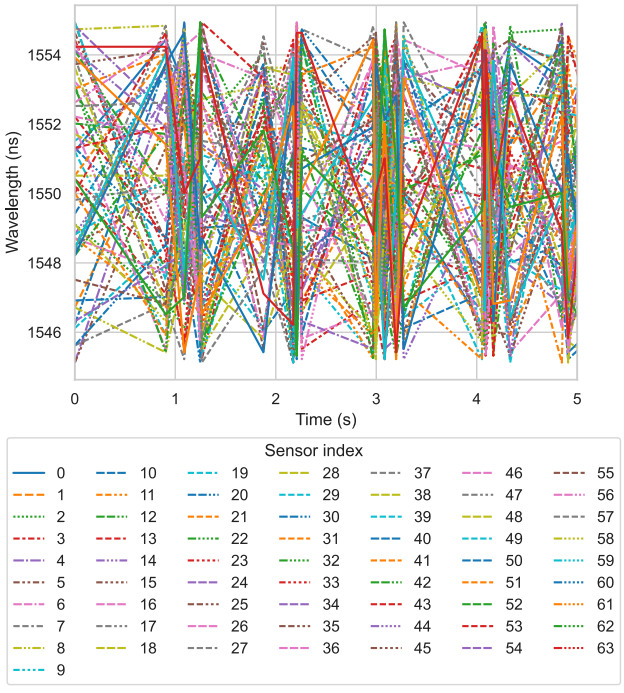

FIGURE 2.10: An illustration of the output generated by the Emulator when tasked with simulating data from 64 sensors linked across 4 interrogator channels. This image was taken from [38].

libraries, are open-source and published under the GNU's Not Unix (GNU) General Public License (GPL) v3.0: their full code is accessible on GitHub [40–43].

2.4.3 Middleware

The Middleware is the core IT component of the project [35, 39]. It is a C/C++ Linux application capable of establishing an Ethernet connection with the interrogator, receiving sensor data, and subsequently transmitting them to the Cloud Database through an Internet connection. As mentioned above, the data transmitted by the interrogator includes further details, including the timestamp, channel, and reticle of each sensor, which helps to understand the current test configuration.
The Middleware incorporates a logging function, which is particularly useful when an Internet connection is unavailable. This precaution prevents data loss by allowing a post-flight comparison between recorded data and those stored in the database to identify any losses during the flight itself. Data that has yet to reach its destination will not be resent. The purpose of this decision is twofold: to avoid confusion between up-to-date and obsolete data during display and to prevent potential saturation

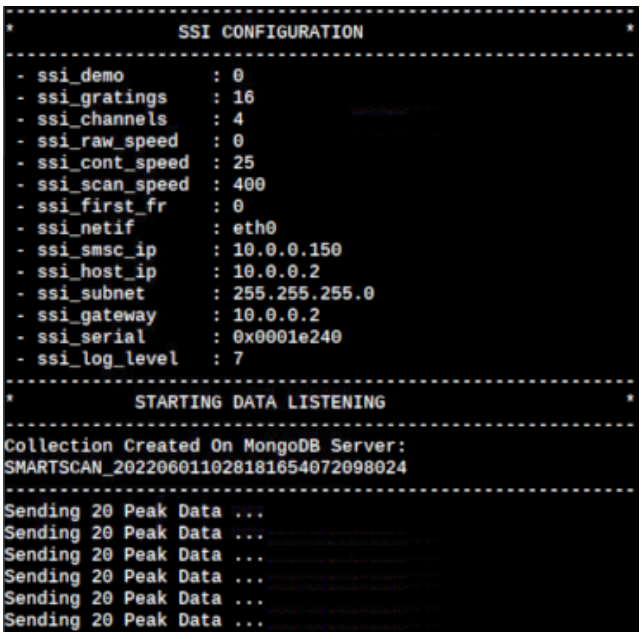

FIGURE 2.11: A screenshot of the Middleware, whose main purpose is constantly receiving data from the SmartScan interrogator or its Emulator, cleaning it, and sending it to the Cloud Database. This image was taken from [35].

of the Internet connection, which must be left free to transmit the most recent data during flights.
Figure 2.11 illustrates the Middleware in action. The software is open-source, and its complete code is accessible on GitHub under the GNU GPL v3.0 [44].

2.4.4 Cloud Database

A Cloud Database allows the possibility of viewing the data anywhere in the world with unlimited viewers. Due to this reason, it was decided to use one for this project. In particular, a Non-Structured Query Language (NoSQL) database eliminates the need to validate incoming data, thereby expediting write operations due to its capability to handle unstructured data.
After meticulously evaluating various contemporary NoSQL databases, MongoDB emerged as the chosen solution [45]. Noteworthy features influencing the selection included the presence of collections and the integration of *Change Stream* technology. In MongoDB terminology, the term *collection* denotes the storage location for similar data records. Conversely, Change Stream is a technology enabling applications to subscribe to all changes in data occurring within a collection, facilitating dynamic reactions based on the received values. This functionality was employed for communication with the subsequent stage of the pipeline, which is constituted by the viewers. In this setup, sensor data is transmitted as soon as it is received by the Middleware.
Upon connection to the Cloud Database, the Middleware establishes a collection, and the name of the newly created collection is defined as *SMARTSCAN_<timestamp>*. Following MongoDB conventions, each individual data record from the Middleware is termed as *document*. These documents are placed within the previously generated collection and saved in Binary JSON (BSON) format [46]. This format, which, as the first word in the acronym indicates, is binary, was developed to address three primary issues associated with the JavaScript Object Notation (JSON) format, particularly relevant when used in a database:

- The limited support for base data types.
- The text-based, human-readable nature of JSON results in slow parsing and inefficient space utilization.

2.4.5 Viewers

The Viewer is the final element of the software application pipeline. Its purpose is to display data from the Cloud Database intuitively. Over time, various viewers have been created: an overview of all the solutions designed so far will be listed below.

2.4.5.1 3D Viewer

The first Viewer ever created for this project, called *3D Viewer*, was realized with Unity [47] and C# to be experienced via common Personal Computers (PCs) [48, 35]. It offers the possibility of representing the data acquired by the interrogator through the following views:

- A *line plot view*, illustrating the variation between the nominal peak reflected wavelength of FBGs and their actual values. As discussed in Section 2.2, these values can change due to mechanical strain and temperature fluctuations, commonly referred to as $\Delta\lambda$.
- A *heat-mapped view*, showing sensor data directly on the 3D model of the UAV through variations in hue and intensity in the vicinity of the physical location of the sensors. The higher the measured $\Delta\lambda$, the greater the deviation from the baseline hue and intensity.

Both representations, each with advantages and drawbacks, enable human-readable monitoring of the state of the UAV and immediate adjustments to its attitude during emergencies. While the heat-mapped view offers more intuitive and easily interpretable information than the line plot view, it is less precise. The users can load any 3D model into the program, facilitating visualization of the heat-mapped for different UAV or diverse scenarios beyond the scope of this article.
Implementing the heat-mapped view on a 3D model presented technical challenges addressed through the use of shaders. They are custom programs designed to run on the Graphics Processing Unit (GPU) pipeline for the purpose of altering the typical processing flow of polygons and images. They were implemented in the 3D Viewer using ShaderLab, a declarative language created by Unity Technologies and comparable to C for Graphics (CG) and High-Level Shader Language (HLSL). Unity automatically converts the code into native languages such as OpenGL Shading

Language (GLSL), HLSL, Vulkan, Metal, and others. This level of abstraction allows the creation of shaders with portability between different GPU manufacturers. The 3D Viewer can connect to the Cloud Database via an Internet connection of any type. If this connection is successful, the 3D Viewer will automatically receive the configuration of the sensors placed on the UAV. In the software configuration phase, these sensors appear as red spheres which can be repositioned on the 3D model by the user through a simple drag-and-drop process in order to accurately replicate their actual positioning on the UAV. This procedure must be carried out with care, as it is essential to have an adequate heat-mapped view. Once the configuration is complete, the sensor data visualization can begin. An image of the initial window of the software is shown in Figure 2.12.
Similar to the Middleware and the Emulator, the entire code of the 3D Viewer is accessible on GitHub under the GNU GPL v3.0 [49].

2.4.5.2 PhotoNext FBG Data Analyzer

The PhotoNext FBG Data Analyzer was developed in C++ using the Qt framework [50, 51]. The main goal behind its creation was to create a more feature-rich and lightweight alternative to the 3D Viewer, although sacrificing the ability to display sensor data through a heat-mapped 3D view. Similar to the 3D Viewer, it allows reading sensor data either through a connection to the Cloud Database,

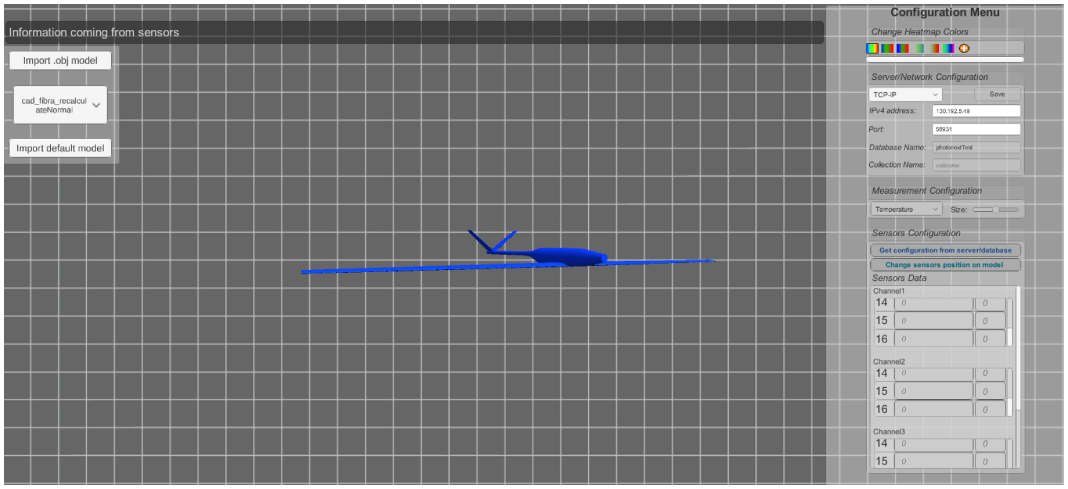

FIGURE 2.12: A screenshot of the 3D Viewer, whose main purpose is to display data from the Cloud Database.

utilizing MongoDB's Change Stream technology, or via a Transmission Control Protocol (TCP) connection with a PhotoNext Middleware application within the local network. The PhotoNext FBG Data Analyzer software comprises three distinct views, accessible through the tab on the right side of the window, with an additional view that can be opened and positioned as needed:

- *Main view*: displays the instantaneous peak wavelength measured by FBG sensors and their trend over time. It is divided into three sections:
  - The left of the window reports a list automatically populated once the connection is established, showing the detected and active sensors. Colored boxes allow the user to choose the display color of each sensor, while checkboxes allow the user to select whether or not to view data from one or more desired sensors.
  - The top-center of the window features a plot showing the instantaneous peak wavelength, with mouse-over displaying corresponding values. A scroll mode allows viewing older data, activated by hovering over the plot and using touchpad/mouse scroll. Returning to live mode is possible by double-clicking the plot or swiping the touchpad/mouse forward.

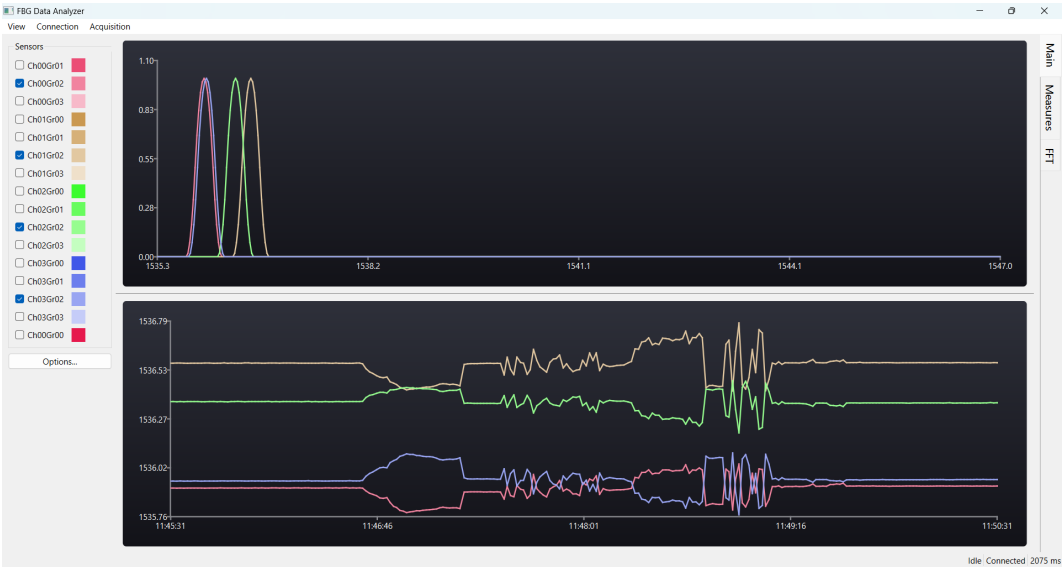

FIGURE 2.13: A screenshot showcasing the main view interface of the PhotoNext FBG Data Analyzer. This image was taken from [51].

  - The bottom-center of the window exhibits a plot depicting the peak wavelength measured by FBG sensors over time. An options button opens a new window for user adjustments, such as turning on/off automatic recalculation of range and grid.

  A picture of the main view is reported in Figure 2.13.

- *Measures view*: similarly to the main view, it shows the temperature or strain measured by the FBG sensors over time, albeit with some key differences:

  - On the left, FBG sensors can be added, and their type (temperature or strain) selected. Proportionality coefficients and starting reference values can be configured accordingly. An optional field for temperature reference is available for reading strain compensation.
  - Due to the impact of both strain and temperature variations on the Bragg wavelength change, the plots of the main view in the PhotoNext FBG Data Analyzer separately display strain and temperature measurements. In the measures view, similar plots are presented, focusing on these specific measurements.

  A picture of the measures view is reported in Figure 2.14.

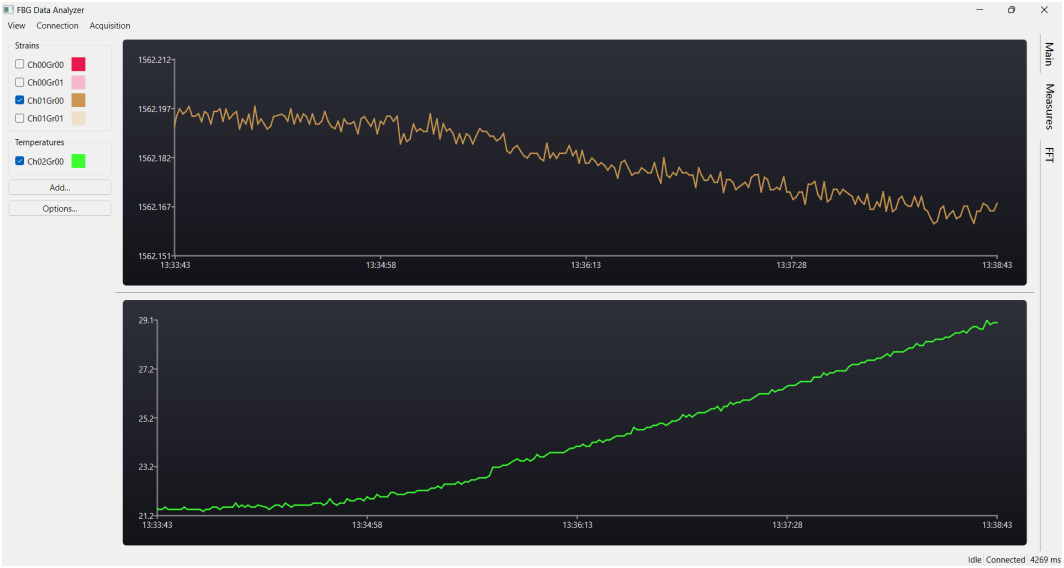

FIGURE 2.14: A screenshot showcasing the measures view interface of the PhotoNext FBG Data Analyzer. This image was taken from [51].

- *FFT view*: enables users to observe the Fast Fourier Transformation (FFT) of selected sensors. After testing, it was decided to calculate the FFT not upon each new sample but upon the arrival of n new samples, aligning with the initial optimization goals for software performance. The options button in this view allows users to modify signal sampling and the number n of samples used for FFT calculation. A picture of the FFT view is reported in Figure 2.15.

The *menu bar*, situated at the top of the window, offers options for establishing connections via TCP on the local network or Cloud Database, tabular data viewing, and logging data locally:

- *Table view*: unlike the other views, it provides an instant, precise visualization of each value detected from the sensors in a separate window. A picture of the table view is reported in Figure 2.16.
- *Connection*: users can choose and configure the type of connection to retrieve data from the Middleware. Options include TCP on the local network or Cloud Database. The connection status (Disconnected, Connecting, or Connected) and the data reception delay are displayed in the bottom bar.

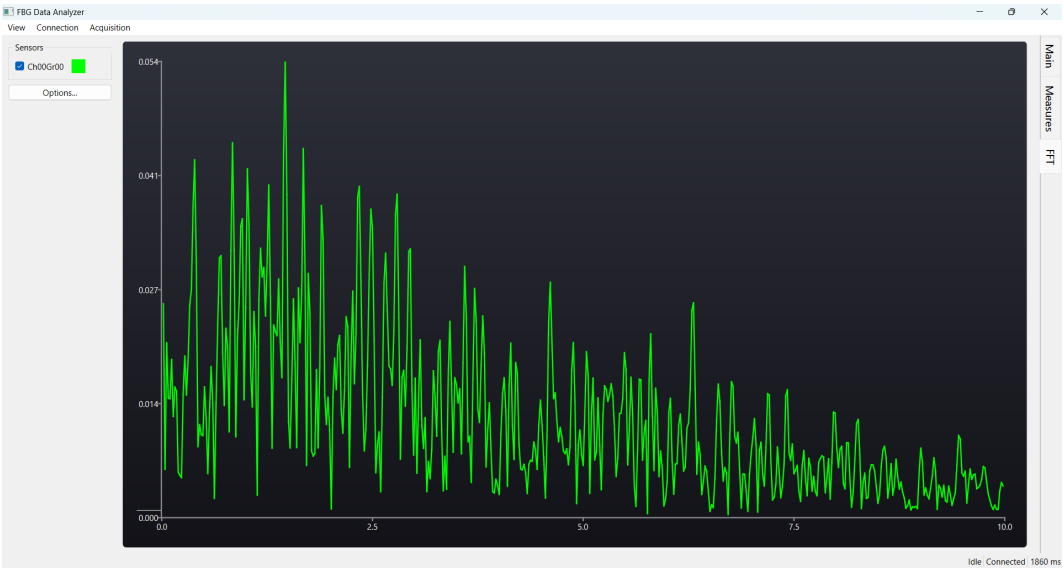

FIGURE 2.15: A screenshot showcasing the FFT view interface of the PhotoNext FBG Data Analyzer. This image was taken from [51].

- *Acquisition*: allows users to configure, start, and stop the logging of sensor data onto the local disk. Parameters like acquisition duration, sampling rate, and the subset of sensors to be logged can be set with the corresponding buttons.

User-configured settings are automatically saved in the settings.json file, which is created when the software is run for the first time. Users can also manually edit settings with a text editor and create backup copies to be used as predefined templates in repeated measurement campaigns. PhotoNext FBG Data Analyzer is distributed as an open-source solution under the GNU GPL v3.0, allowing usage as is or free modification for specific tasks [52].

2.4.5.3 PhotoNext FBG AR Viewer

PhotoNext FBG AR Viewer is an additional viewer that interprets data from FBG sensors in AR. Built with the powerful Unity engine and C# language, PhotoNext FBG AR Viewer is tailor-made for Microsoft HoloLens 2, a cutting-edge AR headset. When the software launches, it first ensures it can connect to the Cloud Database. If there is a connection problem, it will notify the user about the problem itself, such as if the connection failed or if there is no new data to get. If the connection is successful, it will enter the sensor data for the current test and show it in a graph

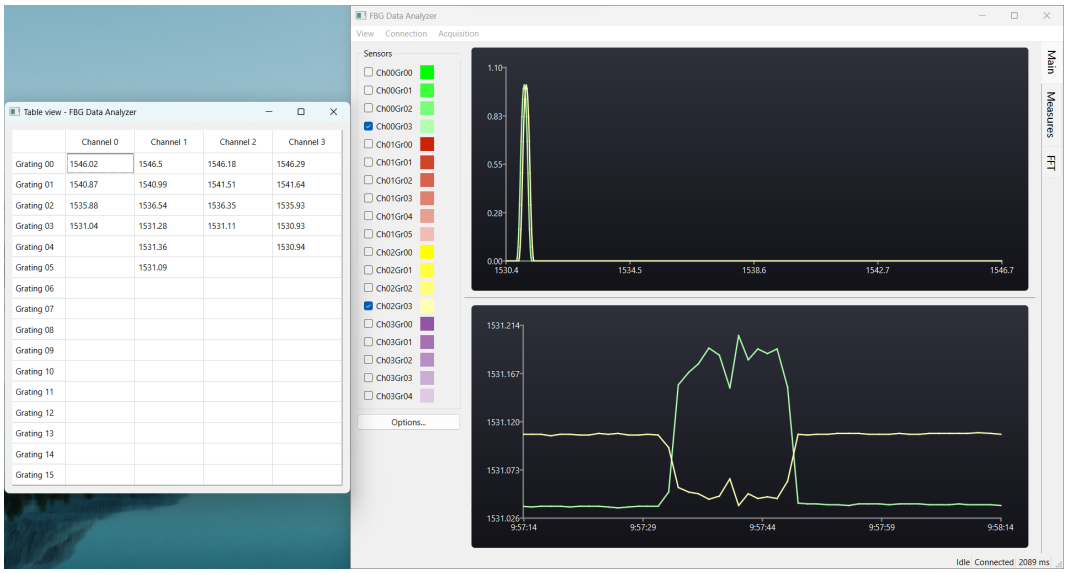

FIGURE 2.16: A screenshot showcasing the table view interface of the PhotoNext FBG Data Analyzer. This image was taken from [51].

illustrating the temporal variation of the wavelength of each sensor relative to its resting value. The software provides buttons for the user to:

- Freeze the position of the line plot on the screen or make it move with its head.
- Show or hide the graph.
- Quit the software.

The software limits the number of times the line plot is updated to improve performance. It then performs a subsampling of the data received. This balance between accuracy and speed ensures the program runs efficiently.

2.4.6 Laboratory Tests

Over time, several laboratory tests were carried out to assess the full functionality of the different elements of the software application chain. They were conducted in an underground laboratory of DIMEAS at Politecnico di Torino. During these tests, the following devices were often used:

FIGURE 2.17: Middleware, Cloud Database, and 3D Viewer Flight Tests - Setup utilized for the strain measurement test, including: the climatic chamber (1), monitored by two screens (2), a laptop running the 3D Viewer application (3), the interrogator (4), the Raspberry Pi (5), with its operations monitored on screen (6), the test sample equipped with the FBG sensor (7). This image was taken from [35].

- A Windows 11 laptop with AMD Ryzen 7 5700U CPU, 32 GB RAM, and 1 TB SSD.
- Alternatively, a Windows 11 laptop with an Intel Core i7-8750H CPU, a NVIDIA GeForce RTX 2080 GPU, 32 GB of RAM, and 6 TB SSD.
- A Raspberry Pi 3 Model B+ was used to run the Middleware.
- The Smart Fibres' SmartScan interrogator was used to read fiber values.
- The carbon-fiber instrumented tail of the Anubi UAV and a multilayer Carbon Fiber Reinforced Polymer (CFRP) were used to perform strain tests.
- The Beger KK-50 CHLT climatic chamber and an uncoated optical fiber sample were used for temperature measurements.
- A Microsoft Hololens 2 was used to execute the PhotoNext FBG AR Viewer.

2.4.6.1 Middleware, Cloud Database, and 3D Viewer Tests

This test focused on the ability of the Middleware to receive data from the interrogator and transmit it to the Cloud Database, enabling near real-time retrieval and visualization by the 3D Viewer.

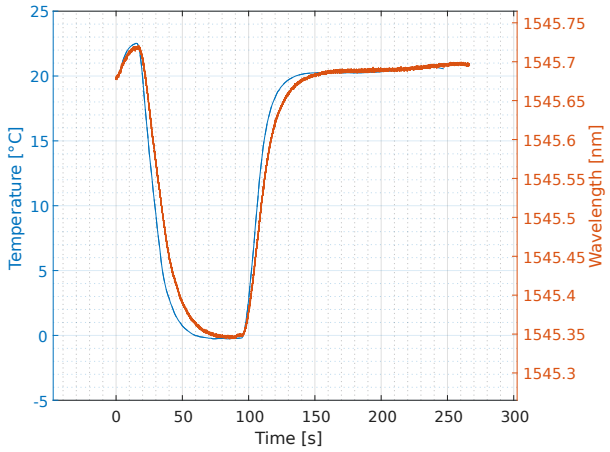

FIGURE 2.18: Middleware, Cloud Database, and 3D Viewer Flight Tests - The wavelength measured by an FBG sensor (depicted in red) and the environmental temperature recorded by a thermometric probe (shown in blue) show a clear direct correlation.

To conduct this test, the system was connected to an uncoated optical fiber sample inside a climatic chamber, allowing variable thermal conditions to be recreated. Figure 2.17 illustrates the configuration used to test the software applications. In FBG sensors, the reflected wavelengths are proportional to the ambient temperature [53]. A preliminary test, shown in Figure 2.18, involved inserting an FBG sensor and a thermometric probe into the climatic chamber to perform a thermal cycle, i.e., a temperature variation that occurs from an initial condition towards the final one, both set by the user. The results indicated a direct correlation between the two measurements: a decrease in temperature corresponded to a decrease in wavelength, while an increase in temperature corresponded to an increase in wavelength.
The effective test was conducted using two separate 4G networks:

1. The first facilitated transmitting data from the Middleware towards the Cloud Database using the aforementioned 4G Internet Key.

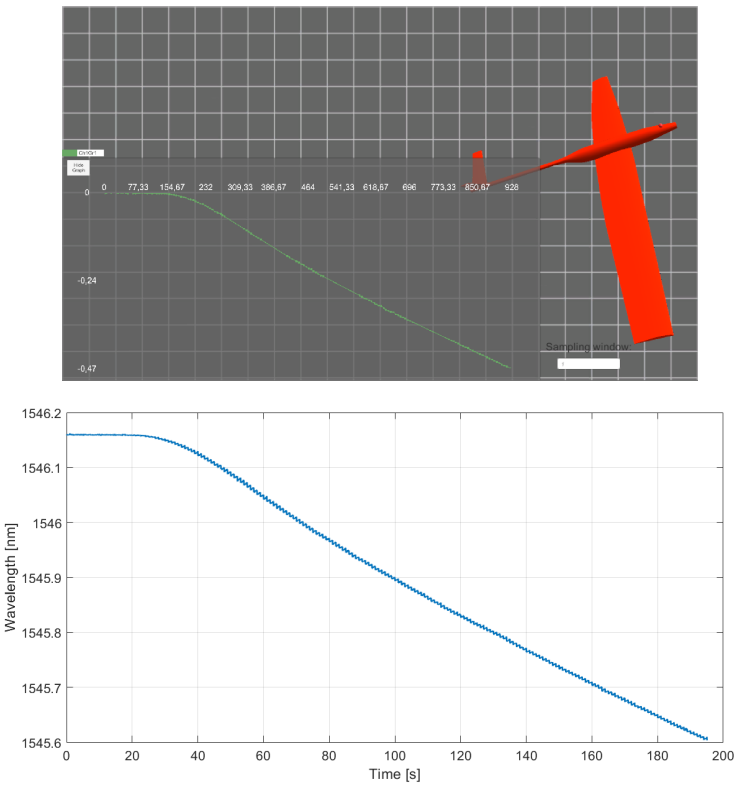

FIGURE 2.19: Middleware, Cloud Database, and 3D Viewer Flight Tests - A comparison between the visualization generated by the 3D Viewer (top) and the corresponding data stored in the Cloud Database (bottom). This image was taken from [51].

2. The second enabled the connection between the Cloud Database and the 3D Viewer, allowing the data to be viewed. For this purpose, the laptop running the 3D Viewer was connected via Wi-Fi to a hotspot created using an Apple mobile phone.

The climatic chamber, which housed the test sample of the optical fiber, was pre-heated to 50°C. Sufficient time was allowed for the fiber to expand and stabilize. Subsequently, the temperature was gradually reduced from 50°C to 10°C in approximately 180s. As expected, the Bragg wavelength varied linearly during the data acquisition period. Figure 2.19 illustrates that the trend displayed by the 3D Viewer closely matches the data stored in the cloud. The latency between the reception of the data by the Middleware and the subsequent visualization by the 3D Viewer ranged between 300 and 800ms.

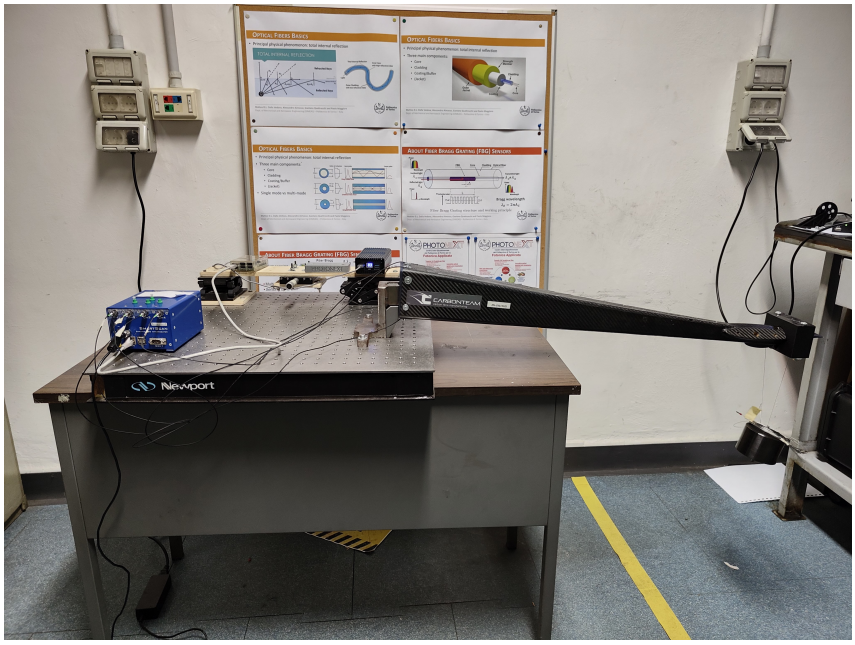

FIGURE 2.20: PhotoNext FBG Data Analyzer Laboratory Tests - Setup utilized for the strain measurement tests, featuring the instrumented tail of Anubi. This image was taken from [51].

2.4.6.2 PhotoNext FBG Data Analyzer Tests

The ability of the PhotoNext FBG Data Analyzer to effectively visualize sensory data was tested under three different conditions: in the first, only deformation was detected; in the second, only temperature was detected; in the third, both were

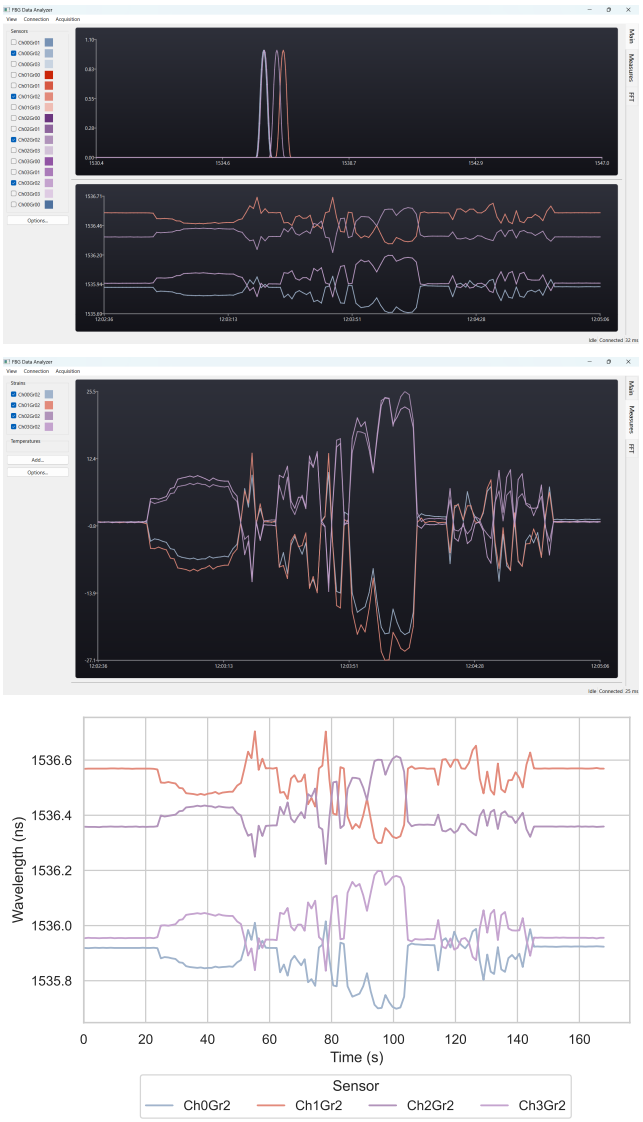

FIGURE 2.21: PhotoNext FBG Data Analyzer Laboratory Tests - A comparison between the Main and Measures views of the PhotoNext FBG Data Analyzer during the strain measurement test (top) and the subsequent analysis conducted on data stored in the Cloud Database (bottom). This image was taken from [51].

detected. The initial test involved measuring strain using optical fibers attached to the carbon fiber tail of Anubi. Since the experiment was conducted in a thermally controlled environment to isolate mechanical loads from temperature variations, the optical variation recorded by the instrumentation was attributed exclusively to the mechanical load applied to the sensor. For this reason, Equation (2.5) can be rewritten as:

$$\frac{\Delta\lambda}{\lambda_B} = K_\varepsilon \Delta\varepsilon \tag{2.8}$$

The tail of Anubi was instrumented with two optical fibers on the upper surface and two on the lower surface. Each optical fiber is equipped with four FBG sensors positioned at variable heights along the tail. To ensure accurate results, the tail was attached to a custom-designed 3D-printed reinforced structure in SolidWorks and secured on a test stand using screws, nuts, and washers. This configuration ensured that externally applied loads effectively stressed the structure and prevented the tail

FIGURE 2.22: PhotoNext FBG Data Analyzer Laboratory Tests - The configuration employed for the temperature measurement test, featuring the climatic chamber housing the network of sensors. This image was taken from [51].

from rotating around its constraint.

Figure 2.21 shows the experimental setup for the strain measurement test. For visual clarity, only 4 of the 16 sensors available were depicted in the test using the PhotoNext FBG Data Analyzer. Using the Measures view of the software, for each of them k_ε was initialized with 0.01 and ε_0 was initialized with 0. A variable load was applied to the end of the tail through a specially designed plastic holder using additive manufacturing. The load acted orthogonal to the tailplane, inducing traction and compression on the sensors. This verified the correct functioning of the software. The test results, shown in Figure 2.21, illustrate how torsional stress on the tail induced compression and tension on several fibers, resulting in wavelength changes detected by the sensors. The optical sensor showed high sensitivity in detecting rapid stress transients, and the developed software effectively captured and graphed the

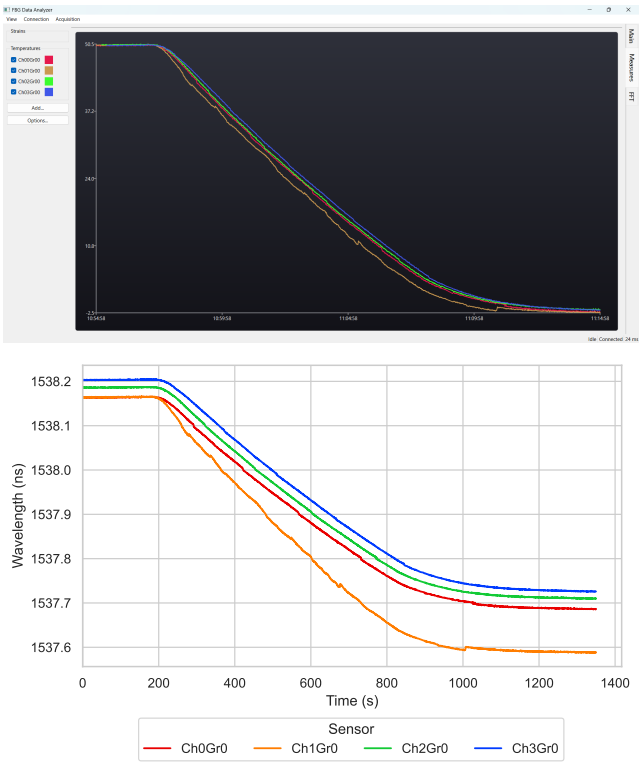

FIGURE 2.23: PhotoNext FBG Data Analyzer Laboratory Tests - A comparison of temperature measurements displayed with the Measures view of the PhotoNext FBG Data Analyzer (top) and the subsequent analysis performed on the data stored in the Cloud Database (bottom). The software can translate wavelength into degrees Celsius and enhance visualization if accurate parameters are provided. This image was taken from [51].

data in near real-time during the testing phases.
The second test involved temperature measurements using 4 FBG sensors positioned inside the Beger KK-50 CHLT climate chamber. The purpose was to evaluate the display of multiple data series on the line plot in the Measures view of the PhotoNext FBG Data Analyzer. The setup for the temperature measurement test is shown in

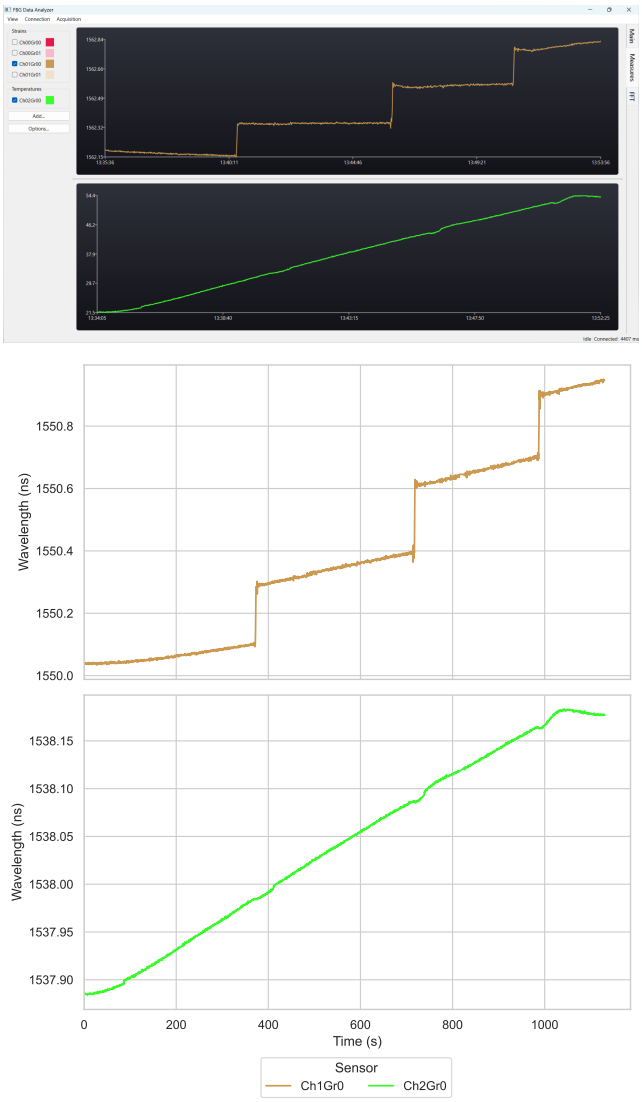

FIGURE 2.24: PhotoNext FBG Data Analyzer Laboratory Tests - A comparison of the simultaneous strain (in ochre) and temperature (in green) measurement displayed with the Measures view of the PhotoNext FBG Data Analyzer (top) and the subsequent analysis performed on the data stored in the Cloud Database (center and bottom). This image was taken from [51].

Figure 2.22. Since only thermal variations are considered significant in this test, Equation (2.5) can be rewritten as:

$$\frac{\Delta\lambda}{\lambda_B} = K_T \Delta T \tag{2.9}$$

Two different instrumented samples, with 4 FBG sensors, were connected to the channels 0 and 2 of the interrogator and placed inside the climatic chamber. Using the Measures view of the PhotoNext FBG Data Analyzer, each was then initialized with a K_T value of 0.01 and an initial temperature T_0 of 25°C. After this step, the climate chamber was preheated to 50°C, giving the sensors time to acclimatize to the new temperature. Subsequently, the temperature inside the chamber was reduced to 0°C, leading to a subsequent decrease in temperature measured by both sensors and accurately displayed by the PhotoNext FBG Data Analyzer software.
The experiment was observed for approximately 20 minutes. Figure 2.23 illustrates the gradual decrease in measured temperatures. While the readings from the three sensors were relatively consistent, the sensor connected to channel 1, represented in Figure 2.23 with an ochre color, showed significantly different values than the others. This discrepancy, reaching differences of about 2°C, has been attributed to using an imprecise Celsius conversion constant.
The latest test involved five different FBG sensors to validate the simultaneous and

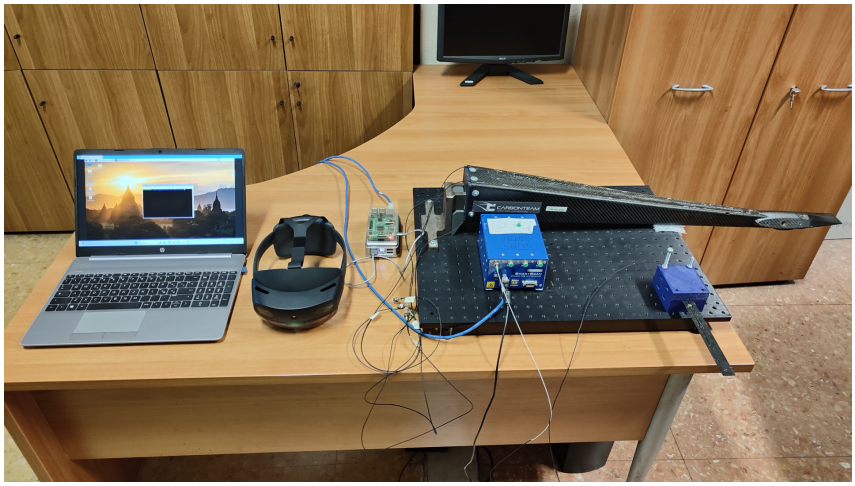

FIGURE 2.25: PhotoNext FBG AR Viewer Laboratory Tests - A picture of the settings used for strain measurement tests, featuring the CFRP, the instrumented tail of Anubi, and the Microsoft Hololens. This image was taken from [54].

effective display of strain and temperature measurements. The affected FBG sensors were connected to the 1 and 2 channels of the interrogator. Using the Measures view of the PhotoNext FBG Data Analyzer, the strain sensors were initialized with a k_ε of 0.01 and an S_0 of 100. In contrast, the temperature sensor was initialized with a k_T of 0.01. and a T_0 of 100. Next, the strain acting on the CFRP sample was measured, while the uncoated optical fiber sample was placed in the climate chamber, set to reach an internal temperature of $100°C$.
For clarity, Figure 2.24 only shows the data from one of the strain sensors depicted in ocher. It can be immediately noticed that it has undergone manual compressions and tractions of varying degrees, while the temperature sensor, shown in green, has recorded an increase in temperature. It can be observed how the system can effectively display both the mechanically applied load, augmented in steps, and the continuous, linear, and constant thermal variation.

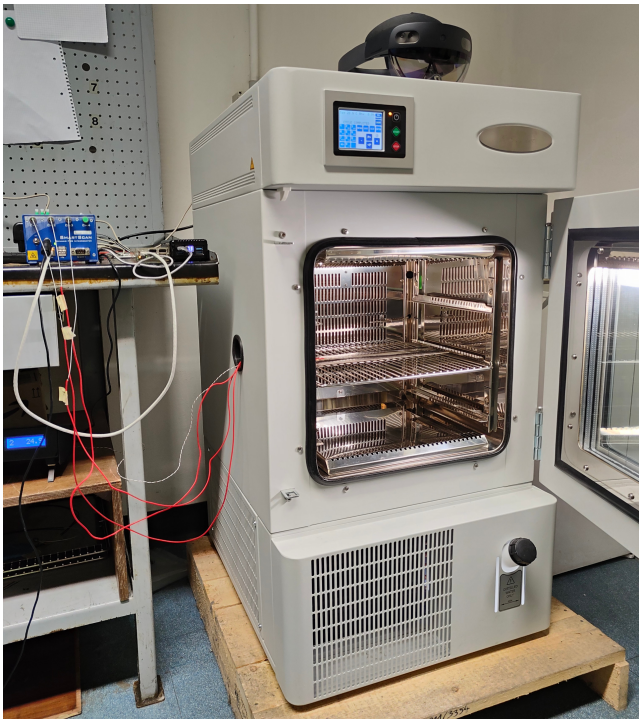

FIGURE 2.26: PhotoNext FBG AR Viewer Laboratory Tests - The configuration employed for the temperature measurement test, featuring the climatic chamber housing the network of sensors and the Microsoft Hololens. This image was taken from [54].

2.4.6.3 PhotoNext FBG AR Viewer Tests

To validate the functionality of the PhotoNext FBG AR Viewer software, a laboratory test campaign was conducted, with the settings depicted in Figures 2.25 and 2.26. The first measurement test focused on measuring the strain experienced by the CFRP. Figure 2.27 illustrates a moment of testing observed through Microsoft HoloLens 2 and the sensor data saved in the Cloud Database. These data were graphed and compared to what was displayed by the PhotoNext FBG AR Viewer to assess its visualization capability. The trend of λ_i over time was explicitly considered. Consequently, the line plot view provided by the software enabled an immediate understanding of the status of the sensor. The subsampling done by the software was evident as the data displayed did not exactly match the saved data. However, the overall trend of the experiment remained comprehensible and constituted an

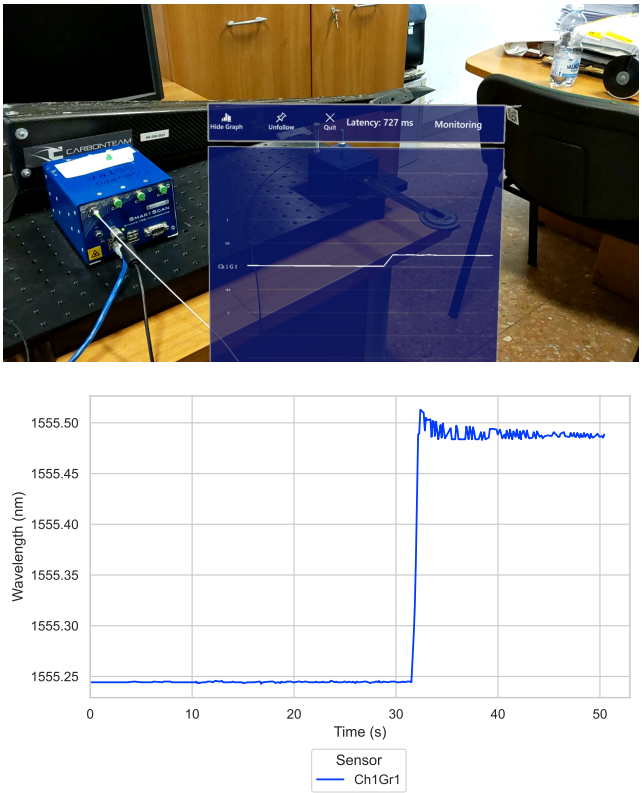

FIGURE 2.27: PhotoNext FBG AR Viewer Laboratory Tests - A comparison between the display given by the PhotoNext FBG Data Analyzer during the first strain measurement test (top) and the subsequent analysis conducted on data stored in the Cloud Database (bottom). This image was taken from [54].

acceptable compromise. In the second test, the strain experienced by the carbon fiber instrumented tail of the Anubi UAVwas measured. Similar to the first test, an external load was applied to the tip of the tail, resulting in traction loads on two fibers and compression loads on the other two. Figure 2.28 compares the data displayed by the software and the data graphed in post-processing from the Cloud Database. Specifically, two sensors were considered: one placed in the tractioned fiber and the other in the compressed fiber. The software effectively facilitated user understanding of the nature of load aging on the sensors by displaying the wavelength trend. Notably, users could easily discern the symmetry of the load induced by the geometry of the tail and the dual disposition of the sensors. Furthermore, the software enabled clear visualization of the individual load steps in the time history, enhancing the user's comprehension of the data provided.

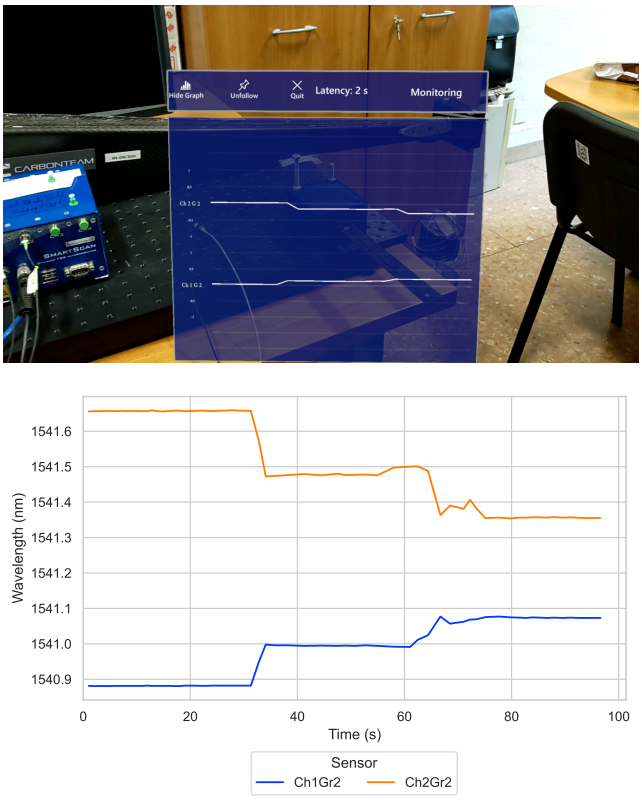

FIGURE 2.28: PhotoNext FBG AR Viewer Laboratory Tests - A comparison between the display given by the PhotoNext FBG Data Analyzer during the second strain measurement test (top) and the subsequent analysis conducted on data stored in the Cloud Database (bottom). This image was taken from [54].

2.4.7 Flight Tests

During the doctoral period, a total of 3 flight test groups were carried out at the Tetti Neirotti airfield, located in the immediate proximity of Turin. In particular, they took place on the following days:

1. December 5th, 2021.
2. October 10th, 2022.
3. July 17th, 2023.

Flight tests were carried out using the Physical System described in Section 2.4.1. For each of them, the technical part was managed by our mixed DAUIN-DIMEAS

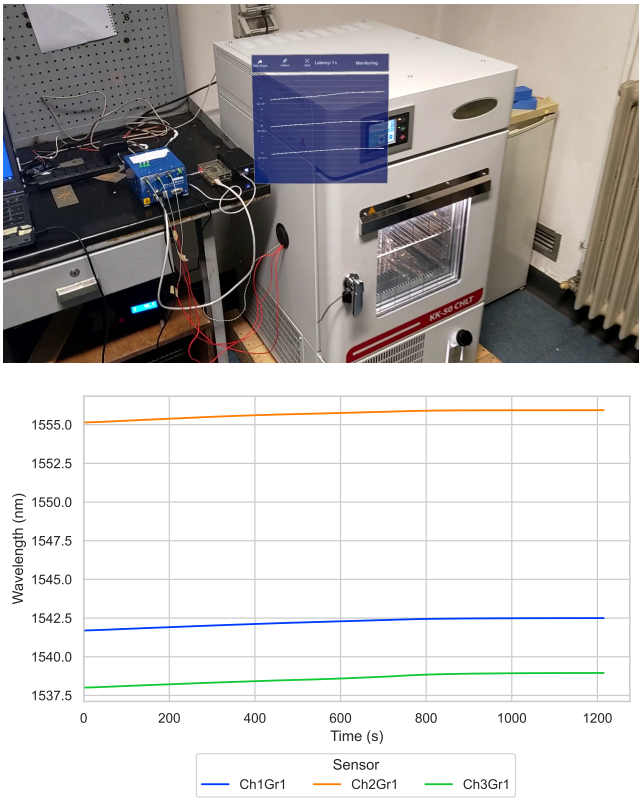

FIGURE 2.29: PhotoNext FBG AR Viewer Laboratory Tests - A comparison between the display given by the PhotoNext FBG Data Analyzer during the temperature measurement test (top) and the subsequent analysis conducted on data stored in the Cloud Database (bottom). This image was taken from [54].

research group, while the ICARUS team handled the assembly and piloting of Anubi. Verification through flight testing is particularly crucial as it allows the software to be subjected to high complexity, thoroughly testing all its functions. They allow for checking the reliability and suitability of the software for practical use. An accurate description of the tests carried out on the various days will be provided below, except for October 10th, 2022, which mainly concerned aeronautical tests.

2.4.7.1 Middleware, Cloud Database, and 3D Viewer Tests

The first flight test group aimed to trial the Middleware, Cloud Database, and 3D Viewer and recreate the real flying test bench. This phase aimed to apply the developed system to a real engineering problem: analyzing the loads acting on the wing of Anubi. The flights took place with three tests lasting between 2 and 10 minutes each.
Figure 2.30 illustrates the experimental setup for these tests. Before thoroughly describing the flight data depicted in Figure 2.31, it is essential to clarify how each FBG sensor produces a reflected wavelength value. For ease of comparison, all sensor data have been normalized using the overall mean value measured by the

FIGURE 2.30: Middleware, Cloud Database, and 3D Viewer Flight Tests - A close-up of the instrumentation installed above Anubi during the three flight test groups conducted at the Tetti Neirotti airfield: SmartScan interrogator (1), Raspberry Pi (2), Internet Key (3), action cam (4). The action cam was utilized to capture graphic documentation of the flights. This image was taken from [35].

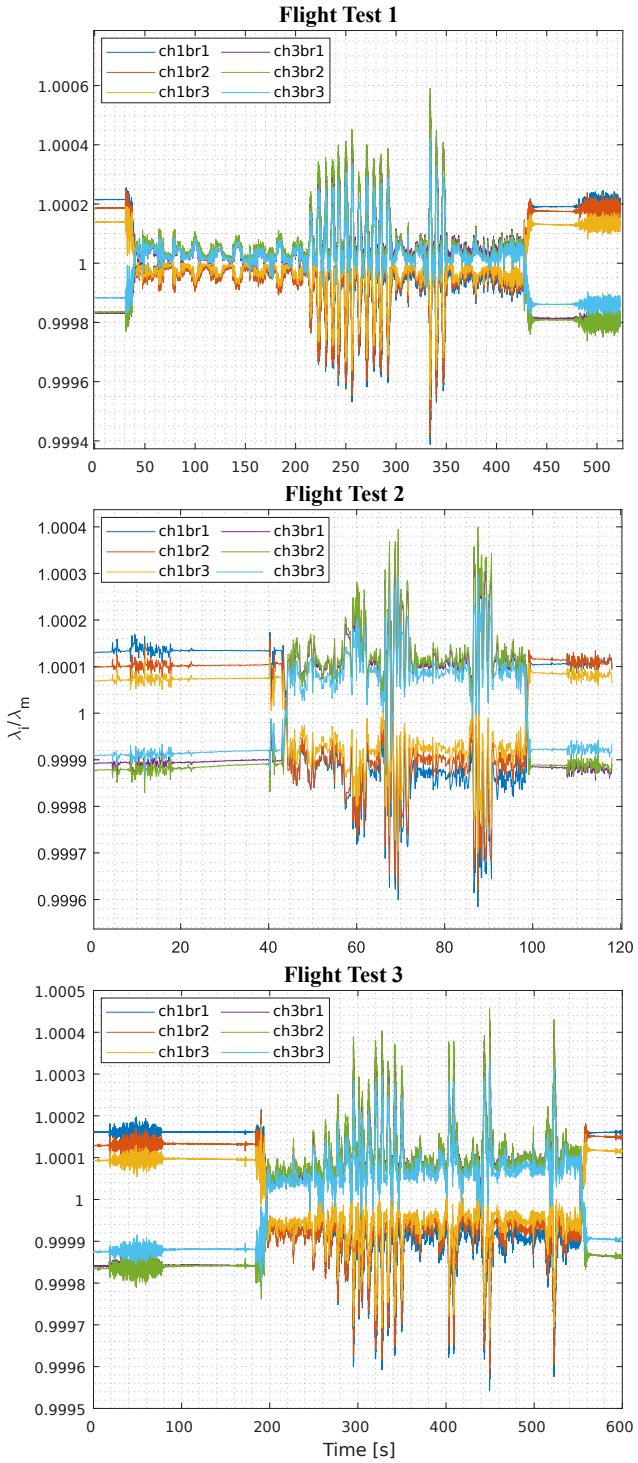

FIGURE 2.31: Middleware, Cloud Database, and 3D Viewer Flight Tests - Trends of the FBG sensors mounted on Anubi during the three test flights. To preserve clarity, only data from six sensors are displayed. This image was taken from [35].

sensors:

$$\lambda_{ig} = \frac{\lambda_i}{\lambda_m} \tag{2.10}$$

where λ_{ig} represents the i^{th} measurement in the plot, λ_i is the i^{th} measurement obtained directly from the data acquisition system, and λ_m is the mean value measured by the specific FBG sensor during the entire flight. In addition, for ease of reading, only six sensors were reported: three on the upper surface of the wing and three on

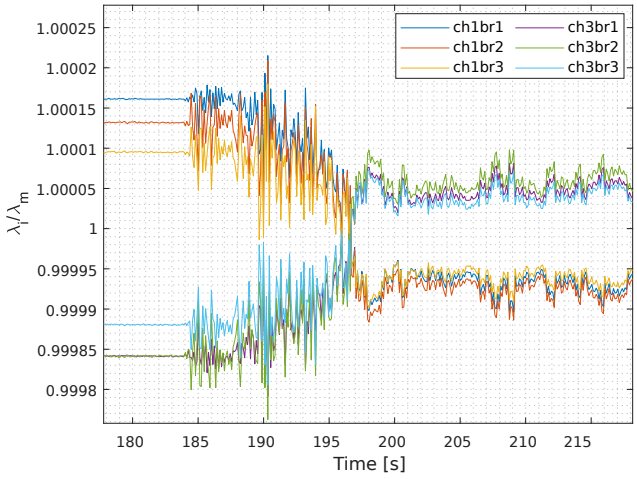

FIGURE 2.32: Middleware, Cloud Database, and 3D Viewer Flight Tests - Notable changes in FBG values during the transition phase from taxiing to flying can be noted in these insights from the third flight of the first flight test group. This image was taken from [35].

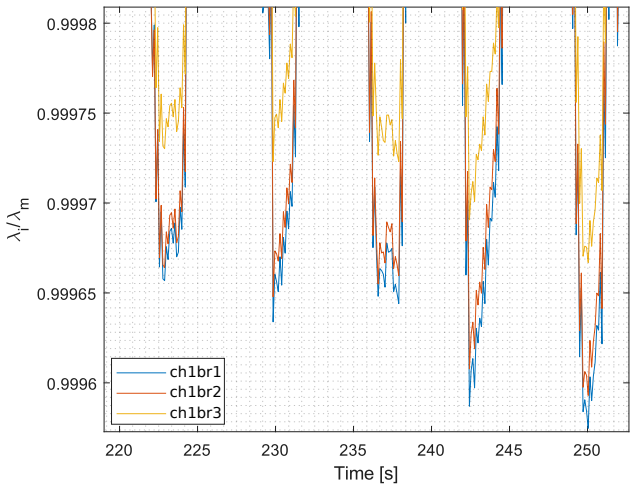

FIGURE 2.33: Middleware, Cloud Database, and 3D Viewer Flight Tests - The different strain levels detected by the FBG sensors during the third flight of the first flight test group. This image was taken from [35].

the lower surface near the wing root. For each of the flights depicted in Figure 2.31, three distinct phases can be clearly seen:

1. A first phase in which the UAV is moved to the airfield. Such data is absent for the first of the three flights.
2. A second, longer phase in which the actual flight occurs.
3. A third phase in which the UAV is recovered from the airfield.

As regards the actual flight, three different stages can also be noted:

1. *Taxiing*: The wing is influenced solely by its weight during this stage. Stable or slightly vibrating data due to the ground movement of the UAV can be observed. At the moment of takeoff, due to aerodynamic forces, the upper surface of the wing experiences compression during flight, while the lower surface undergoes traction. This phenomenon is evident in the plots, where sensors experiencing traction increase in wavelength values while sensors experiencing compression show a decrease. These trends align precisely with the stress distribution on the wing as detected by the FBG sensors.

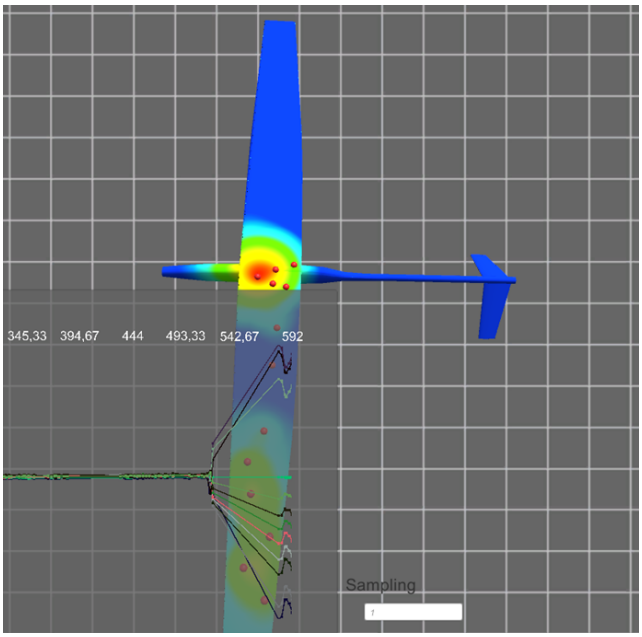

FIGURE 2.34: Middleware, Cloud Database, and 3D Viewer Flight Tests - A close-up of the 3D Viewer interface during one flight, showcasing the heat-mapped and line plot views. This image was taken from [35].

2. *Flight*: In this stage, it is possible to notice significant changes in reflected wavelength due to the aerodynamic loads acting on the UAV. Each peak corresponds to a specific maneuver performed by the remote pilot. These observations are interesting because FBG sensors can detect the vibrations of the wing and the loads experienced during maneuvers.

3. *Landing*: During this stage, it is clearly visible the moment of impact with the ground where, unlike taxiing, the upper surface of the wing undergoes traction, while the lower surface undergoes compression. Stable or slightly vibrating data follow due to the ground movement of the UAV.

Figure 2.32, which depicts a segment from the third flight, it is possible to see the deflection of the wing during takeoff due to aerodynamic forces. This deflection results in opposite readings from the sensors: those on the upper surface, connected to interrogator channel 1 (ch1), record compressive loads, while those on the lower surface, connected to interrogator channel 3 (ch3), detect traction loads. As a result, the curves that initially exceed the value 1 decrease below this threshold and vice versa. This pattern is repeated during the landing phase.
From the detail of the third flight shown in Figure 2.33, it is worth noting the varying levels of induced strain detected by FBGs positioned along the same line. This discrepancy arises because the sensors are mounted at different distances from the root of the wing and highlight the remarkable sensitivity of the developed system.
Regarding the IT aspect, a higher latency was observed during the flight test group

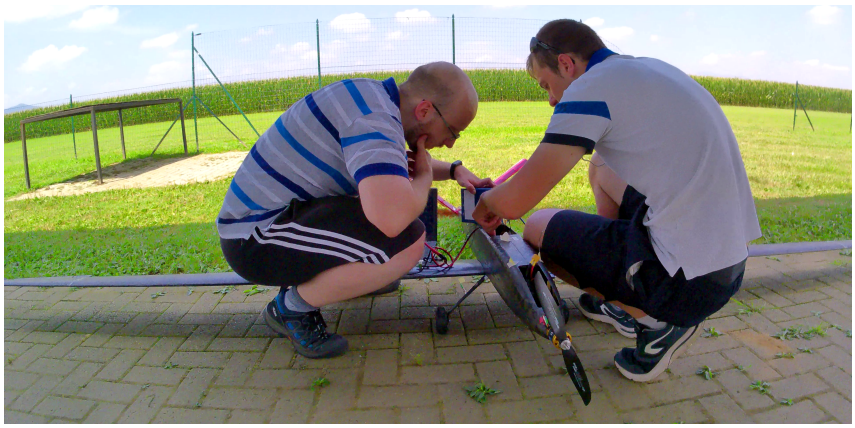

FIGURE 2.35: FBG Data Analyzer Flight Tests - The assembly procedures for the third flight test group stand. This image was taken from [51].

than in the laboratory tests, with values ranging between 800 and 2000ms. This was probably the result of the rapid changes in position, attitude, and speed of the aircraft and the sub-optimal network signal strength. In contrast, laboratory tests were conducted under controlled conditions, benefiting from excellent network signal strength. Nevertheless, a 100% success rate in sending data was observed in all cases.

In conclusion, this flight test group confirmed the proper functioning of the software applications chain. Figure 2.34 shows a screenshot of the 3D Viewer illustrating the Bragg wavelength variation for all sensors via the two views offered by the software.

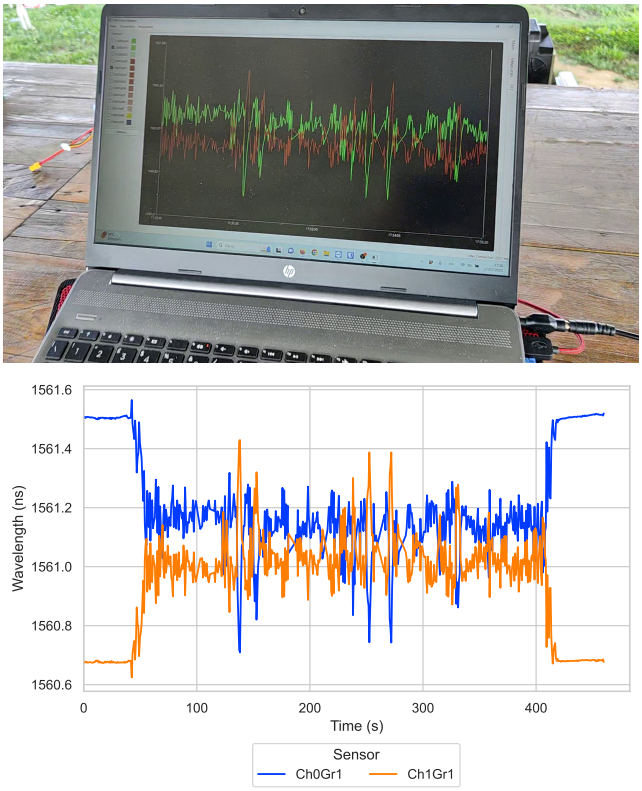

FIGURE 2.36: FBG Data Analyzer Flight Tests - Comparison of near real-time data visualization during the third flight test group using the PhotoNext FBG Data Analyzer with post-processing analysis conducted on data stored in the Cloud Database. This image was taken from [51].

2.4.7.2 FBG Data Analyzer Test

The third and final flight test group carried out so far concerned the verification of the PhotoNext FBG Data Analyzer software. Figure 2.35 reports the flight test stand setup. In contrast, Figure 2.36 shows the graphical output provided by the PhotoNext FBG Data Analyzer during one of the flights performed during the test group. The observed trends align correctly with the graphs representing the data saved in the database and generated after the conclusion of the test. Figure 2.37 shows the data obtained during the other two flights, further validating the usefulness of the software and the features offered.
This flight test group, like the previous ones, confirmed the robustness of the developed software, demonstrating its effectiveness in achieving the objectives of the research group.

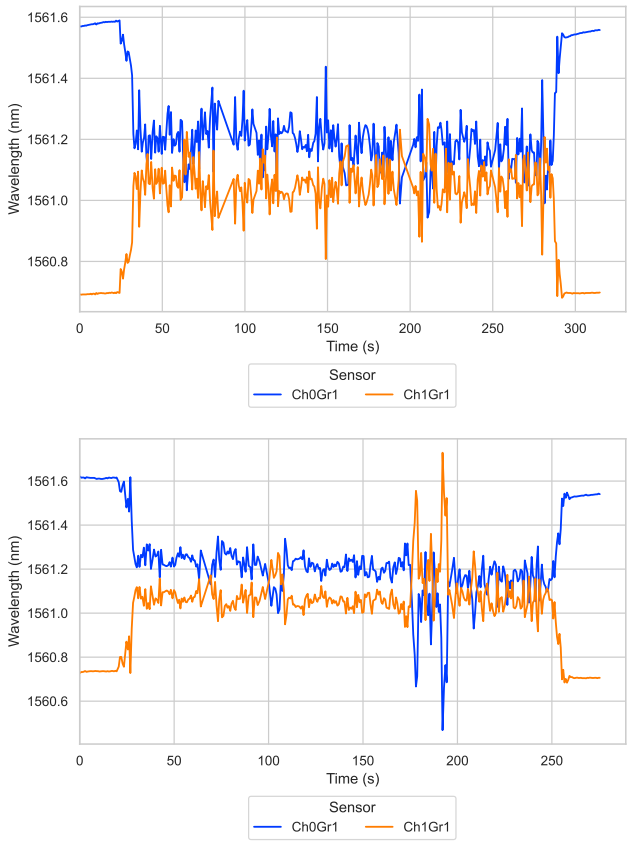

FIGURE 2.37: FBG Data Analyzer Flight Tests - Data captured during the other two test flights carried out during the third flight test group.

Chapter 3

Image Sensors

This chapter presents projects involving the use of image sensors in order to address two different problems. The first relates to countering the *Coronavirus Disease 2019 (COVID-19)* pandemic by identifying the type and manner in which masks and respirators are worn. In contrast, the second relates to *waste management*, mainly focusing on improving recycling quality.

3.1 Image Capturing Devices

Personal photography is made possible by many types of devices, such as camera phones, Digital Still Cameras (DSCs), and camcorders [55]. In addition to cameras for personal use, there are also cameras designed for specific applications, such as automobiles, security systems, and medical endoscopes. Furthermore, the television industry relies heavily on broadcasting cameras that deliver high-definition images. Cameras are not limited to capturing visible light; they are also used for thermog-

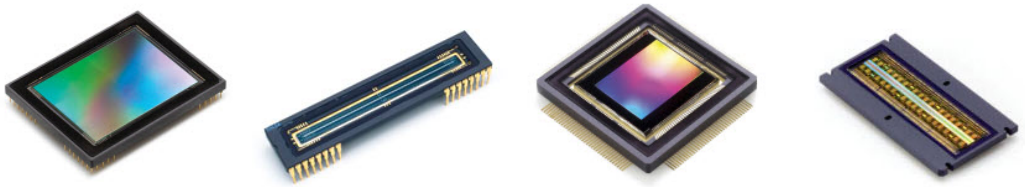

FIGURE 3.1: CCD sensors (left and center-left) vs CMOS sensors (center-right and right). This image was taken from [56].

raphy, which visualizes heat distribution through infrared imaging and X-ray and ultraviolet imaging. Some cameras can capture extremely high-speed phenomena, while others are designed to obtain highly accurate color information. There are even cameras used exclusively by machines, such as those for automated driving and machine vision, which analyze and interpret image data. In short, cameras are utilized in a vast array of fields. Nevertheless, why are there so many different types of imaging systems, with cameras being a prime example? The answer is that each imaging system is designed to capture images with the appropriate level of quality for its specific purpose. Within each system, the role of the image sensor is to collect image information of sufficient quality for that particular application.

3.2 Image Sensor

An image sensor, also called an optoelectronic sensor, is a grid of optoelectronic elements, called pixels, typically arranged in rows and columns in a similar way to display pixels even if they perform the opposite function, since differently from these, they convert an optical signal into one electrical signal [55] [58]. The pixels of the image sensor are tiny as they are on the order of micrometers, and the rectangular area of the sensor, roughly the size of a postage stamp, can contain several million pixels. The size of the sensor and the density of its pixels are crucial factors that determine several essential parameters related to the quality of the captured image. For example, the acquisition time is related to the electronic structure of the sensor. A photographic camera may require a minimal acquisition time to capture clear images of moving

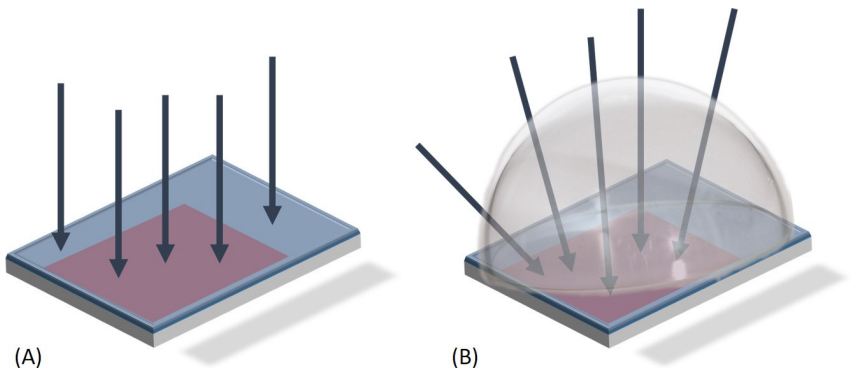

FIGURE 3.2: Increase in FF following the application of microlenses. In the illustrated case, there is a transition from 60% (A) to 100% (B). This image was taken from [57].

objects. In contrast, a video camera must accelerate data transfer to enable the sensor to capture new images at a rate suitable for obtaining video. Cameras primarily use two main types of image sensors: *Complementary Metal-Oxide Semiconductor (CMOS)* and *Charge-Coupled Devices (CCD)* structures. Figure 3.1 shows some examples of CMOS and CCD sensors.

3.2.1 Characteristics of Image Sensors

Image sensors possess several key attributes that are crucial in determining their quality, applications, and overall performance. This Section focuses on two of the most significant characteristics of these sensors: the *Fill Factor (FF)* and the *Full Well Capacity (FWC)*. Understanding these features can provide valuable insights into the capabilities and limitations of various image sensors.

3.2.1.1 Fill Factor

The FF refers to the proportional relationship between the area sensitive to light and the total area of a pixel [57]. A higher FF suggests a more extensive active sensor surface. FF values for CCD are approximately 90%, while CMOS have values around 30% due to their control circuitry. Initially, the low FF of CMOS sensors posed a significant limitation. However, progress in optomechanical technology has facilitated the creation of converging lenses with diameters spanning merely a few tens of micrometers. These lenses, called microlenses, can guide incoming light rays

FIGURE 3.3: Having a larger FWC leads to an increase in dynamics. The image on the left is taken with a 16-bit dynamic, while those on the right have a lower dynamic. The galactic center is completely burned out, as are the background stars in the case of the narrow dynamics image. This image was taken from [57].

toward the sensitive regions of the pixel, thus boosting the FF to near-100% values. When making a purchase, it is essential to ask about the actual pixel FF rather than the microlens-adjusted one. Microlenses enhance the FF and minimize the halo effect resulting from light scattering off the non-photosensitive pixel components. This is especially significant when dealing with intense light sources. Figure 3.2 reports an example of applying microlenses to increase the FF.

3.2.1.2 Full Well Capacity

FWC refers to the maximum quantity of electrons that a photosensitive element can accumulate [57] [59]. This attribute is reliant on the capacitive capability of the photoelement, which can be visualized as a well containing photoelectrons. The FWC generally varies between tens and hundreds of thousands of electrons, determining the highest number of photons interacting with a photoelement before it reaches saturation. Figure 3.3 shows an example of images taken with two different FWC. A specialized control circuitry, known as antiblooming, is commonly incorporated to prevent the charge from overflowing into neighboring pixels, serving as an electronic shutter. The FWC is influenced by numerous factors, including FF, intrinsic pixel architecture, depletion region size, and more. Figure 3.4 shows the relation between FWC and pixel size.

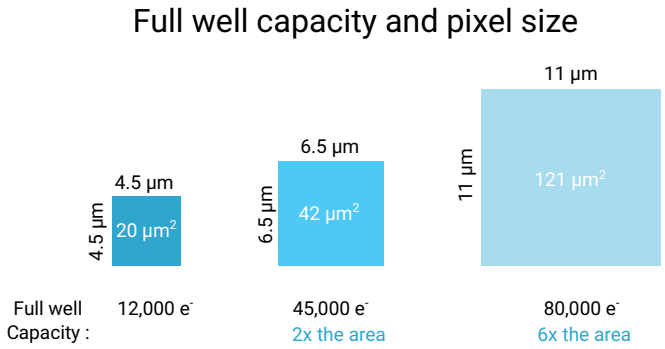

FIGURE 3.4: The area occupied by the FWC for different cameras with 4.5, 6.5, and 11 μm pixels. As the pixel size increases, the FWC also increases. This image was taken from [57].

3.2.1.3 Mono and Color Sensors

As shown in Figure 3.5, there are two primary categories of visible light sensors: color and monochrome (mono) sensors [55].
Color sensors have an additional layer called a color filter positioned beneath the microlens. The color filter absorbs unwanted color wavelengths, ensuring that each pixel is sensitive to a specific color wavelength. The most common color filter arrangement is the Bayer filter, which consists of a pattern of red, green, and blue filters. This results in an image where each pixel captures information for one of the primary colors. Through a process called demosaicing, the full-color image is reconstructed by interpolating the color information from neighboring pixels.
On the other hand, monochrome sensors have no color filter. Consequently, each pixel is sensitive to all visible light wavelengths. This versatile feature allows monochrome sensors to capture more light and provide higher sensitivity than color sensors. The absence of a color filter also eliminates the need for demosaicing, resulting in higher resolution and lower noise levels. Monochrome sensors are often used

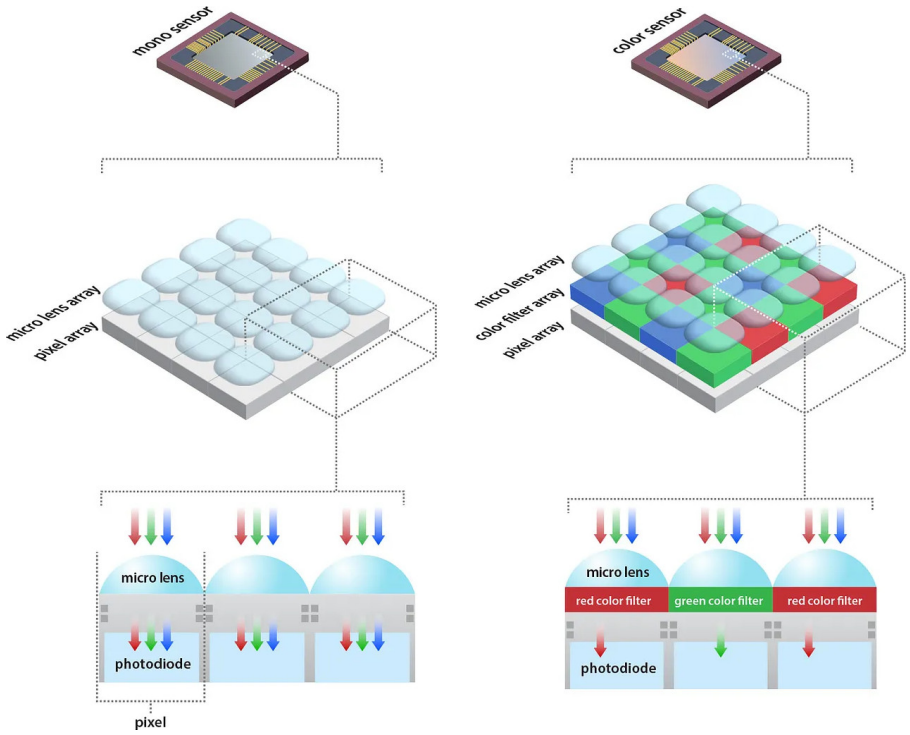

FIGURE 3.5: The images provided show the sensor planes for both monochrome (left) and color (right) sensors with a Bayer pattern. This image was taken from [60].

in applications where color information is not essential, such as low-light imaging, machine vision, and scientific research.
The choice between color and monochrome sensors depends on the specific requirements of the application, with color sensors being suitable for capturing full-color images and monochrome sensors offering better sensitivity and resolution for applications where color information is not necessary.

3.2.2 CMOS

CMOS is a technology used in digital electronics to design integrated circuits whose main component is the Metal-Oxide Semiconductor Field-Effect Transistor (MOSFET) transistor inverter [55] [58]. CMOS circuits, shown in Figure 3.6, consist of a pull-up network, which operates the high logic level (1), and a pull-down network, which operates the low logic level (0). In CMOS image sensors, the conversion of the light signal into an electrical one occurs directly inside the chip/sensor. Each photodiode in a CMOS sensor has its amplifier and Analog to Digital Converter (ADC). This on-chip circuitry reduces the FF, which, as seen above, is the ratio of the light-sensitive area to the total area of the pixel. As a result, using microlenses becomes essential to increase the light-gathering capability of each pixel. Microlenses

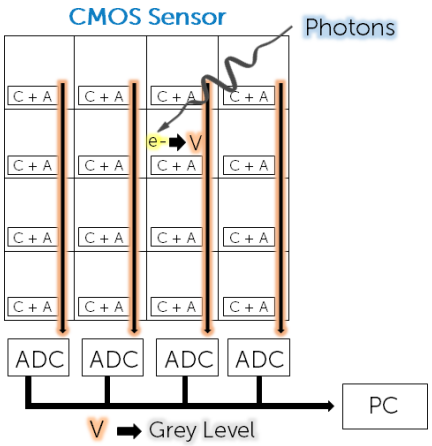

FIGURE 3.6: In CMOS sensors, photons strike the pixels, are converted into electrons, and subsequently converted into voltage within the pixel itself. Each column of pixels is then read out independently by dedicated ADCs and sent to a PC for display. This image was taken from [61].

focus the incident light onto the light-sensitive area of each photodiode, improving the sensitivity and overall performance of the sensor.

3.2.3 CCD

CCD sensor, represented in Figure 3.7, is an integrated circuit consisting of a row or grid of semiconductor elements called photosites [55] [58]. These elements can accumulate an electric charge proportional to the intensity of the electromagnetic radiation that hits them. When stimulated by an electrical impulse, the photosites are paired so that each can transfer its charge to another adjacent element. In a CCD sensor, converting the light level into digital data takes place outside the sensor with a dedicated chip.
One of the drawbacks of CCD sensors is a defect known as Residual Bulk Image (RBI) or ghost effect, especially at low temperatures. This occurs when not all photoelectrons tend to abandon the potential hole during readout, causing an overlap between the previous and new images. The sensor can be irradiated with Near Infrared (NIR) radiation to reduce this problem and create a smeared ghost effect over the entire image, making it less evident. Another problem that can affect some CCD sensors is smearing, a phenomenon that occurs when, during the readout phase or frame shift, which is the process of transferring an image from the sensor to

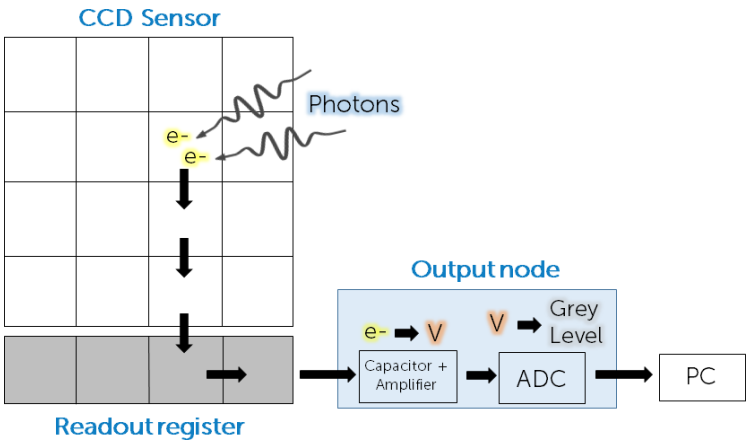

FIGURE 3.7: In CCD sensors, photons strike the pixels, are converted into electrons, and are then transferred down the sensor to the readout register. The readout register sequentially shifts the electrons to the output node, where they are first converted into a voltage signal and then into grey levels, representing the intensity of the captured light. The processed digital data is then transmitted to a PC for display. This image was taken from [61].

storage, photons continue to impact the pixels. This leads to unusual effects, such as distorted helicopter blades, among other visual artifacts.
CCD sensors have been widely used in various applications, including digital cameras, astronomical telescopes, and medical imaging devices. However, they have mainly been replaced by CMOS sensors in many consumer applications due to their ability to integrate more functions on a single chip, lower power consumption, and cost. Nevertheless, CCD sensors are still used in high-end applications where high sensitivity and low noise are critical. The CCD sensors can be classified as follows:

- *Full Frame CCDs*: in this type of CCD sensor, the scanning operation is carried out progressively, i.e., in a single shooting period (integration), a whole frame is captured [55] [58]. They do not have a pixel storage area, and during readout, smearing may occur. A mechanical shutter or stroboscopic light may be required to prevent this problem. Full Frame CCDs have a FF of 100%.
- *Frame Transfer Sensor*: it has a data storage sensor located below the active sensor [55] [58]. It is shielded and capable of receiving data in less than 500 microseconds, but at the cost of an increased price. Smearing is reduced, and the FF is excellent, reaching 100%.
- *Interline Transfer Sensor*: it is the most used type of CCD sensor today [55] [58]. Each pixel has a storage area next to it, capable of receiving data in 1

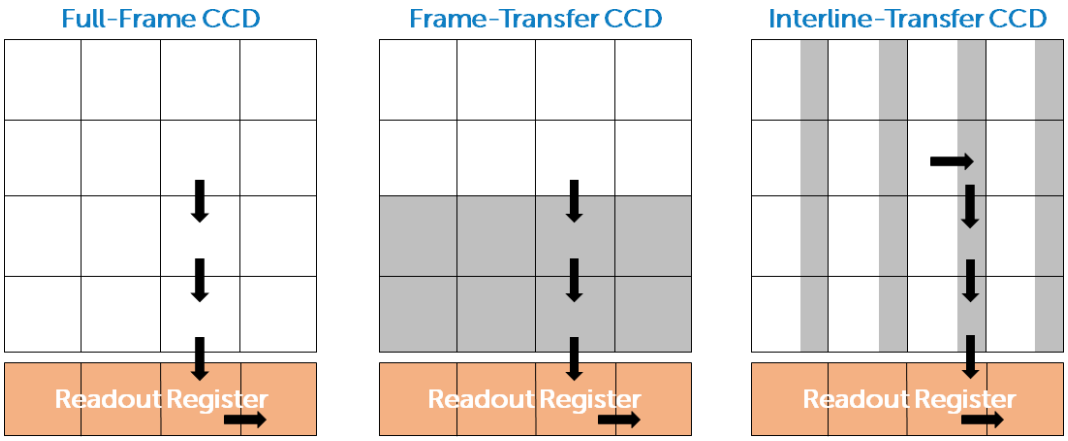

FIGURE 3.8: There are several CCD sensors, each with its own design and functionality. Gray areas represent masked regions that are not exposed to light. The line-transfer sensor has a section of each pixel masked (gray), while the frame-transfer sensor has an active image array (white) and a masked memory array (gray). This image was taken from [61].

microsecond. This area is adequately masked so that photons cannot hit it. The storage areas of pixels form a vertical column at the bottom of the sensor. It does not present smearing but has a FF 20% lower than other types. It requires neither a shutter nor a strobe light.

Figure 3.8 graphically represents the differences between the different types of sensors CCD mentioned above.

3.2.4 CMOS vs CCD

Here is a summary of the advantages and disadvantages of CMOS and CCD sensors [55] [58]:

- Advantages and disadvantages of CMOS:
  - + Lower power consumption.
  - + Less expensive.
  - + More straightforward design.
  - − Lower image quality.
  - − More susceptible to noise.
- Advantages and disadvantages of CCD:
  - + Higher image quality.
  - + Lower noise levels.
  - − Higher power consumption.
  - − More expensive.
  - − Greater complexity.

Over time, both CCD and CMOS technologies have been improved. For CMOS, the focus has been on improving image quality, while for CCD, the focus has been on reducing power consumption. As a result, the two sensors are comparable in terms of performance and can be used interchangeably in most applications. However, in some specific applications, one type of sensor may still be preferable to another based on specific requirements.

3.3 COVID-19 Pandemic

The COVID-19 pandemic has drastically changed our lifestyles and the perception of certain behaviors. In many countries, the situation has now returned to normal, but only two years ago, the situation was very different from the current one. The tactic most used by states to minimize the risk of contagion was to implement restrictions on personal freedom, impose masks in public places, and avoid crowds. Despite this, many individuals have engaged in incorrect and harmful behavior, jeopardizing the progress in containing the virus. Instances such as crowds during console releases, lockdown fears, or Black Friday and Cyber Monday Christmas offers serve as clear examples of this issue [62–65].
During the most intense period, researchers and companies from various fields worked hard to create solutions to combat its spread. In this regard, our research group has also made contributions. In particular, work has been done on creating databases for mask detection systems capable of automatically identifying whether a person is wearing a mask or a respirator. Before the COVID-19 pandemic, these databases were limited and primarily used to monitor mask usage in highly polluted areas [66] or construct automated safety systems [67]. The arrival of the pandemic has led to a significant increase in their number. The contributions made will be discussed in the following Sections.

3.3.1 Facial Mask and Respirator Database

The first effort of the research group against the COVID-19 pandemic was represented by the Facial Mask and Respirator Database (FMR-DB) [68, 69]. Unlike other databases released in the same period, FMR-DB offers accurate classification of the essential types of facial masks and respirators, along with their degree of protection. Sometimes, masks can have very fine details that can make the difference: an example of this is shown in Figure 3.9. Released in an open-access format, FMR-DB is readily available to the scientific community for further research and development. A proof-of-concept for a mask detection system using EfficientNet-B0 [70], a state-of-the-art Convolutional Neural Network (CNN), was released along with the database. The primary objective of this system was to implement automatic controls for entry into high-security areas, ensuring the safety and comfort of individuals within these spaces. This technology was not only valuable for helping mitigate the spread of the

COVID-19 but also for protecting people working in environments with particles, chemicals, or other impurities, such as metalworking industries, lumber mills, or facilities handling radioactive materials, including nuclear power plants.

3.3.1.1 Database Creation

The FMR-DB comprises 2565 images from the Internet and is freely available in an open-access format. While the number of images may seem relatively modest, it is essential to note that quantity alone does not necessarily correlate with system quality; classification accuracy is equally crucial. Given the urgency of the COVID-19 crisis, a balance was sought between these two factors. Image search was conducted using different search engines such as Google Images, Bing, and DuckDuckGo. Since these platforms employ different search algorithms, they yield diverse results, increasing the likelihood of discovering new images. Care was taken to ensure that each class contained an equal number of images to prevent potentially detrimental imbalances during system implementation. Due to their diverse sources, the images vary in resolution and quality.

FIGURE 3.9: Respirators with and without valves are profoundly different. The valve helps protect from viruses and bacteria and offers better breathing, but it cannot be used to protect others from infection as it makes air escape easier. On the contrary, respirators without valve can be used equally for personal protection or the protection of others. Valves are usually small and variable in shape and may have a different physical location, making it essential for a mask detection system to accurately identify their presence or absence. This image was taken from [68].

3.3.1.2 Database Characteristics

During the design of the FMR-DB, the aim was to create a versatile image classification system. The dataset is divided into two main categories: images of people wearing face masks or respirators and images of people without them. These categories will be discussed in detail in the following Sections.
The FMR-DB is available in two versions: one with the original images obtained from the Internet and another with only the faces of the people portrayed, manually cropped to ensure uniformity and to avoid cutting off any part of the respiratory protection or other facial features. Although automatic face detection algorithms like Viola-Jones [71] or Multi-Task Cascaded Convolutional Neural Networks (MTCNNs) [72] can be used, manual cropping was chosen for better control over the process. Most images were cropped into a square shape with the faces centered as much as possible. In cases where this was not feasible, a rectangular crop was used. Users may add padding or extend the images to a new dimension, but no padding was added for the tests conducted in this study. Each image is labeled with a name consisting of a series of tags denoting its characteristics, followed by a unique number. The list of labels and their meanings can be seen in Table 3.1.

3.3.1.3 Facial Masks and Respirators Classification

Images containing facial masks or respirators are categorized based on their type, which can be one of the following:

- Disposable Respirators With Valve.
- Disposable Respirators Without Valve.
- Full-Face Respirators.
- Half-Face Respirators.
- Surgical Masks.
- Non-Medical Masks.

Each class contains 315 images and has an internal classification based on the protective clothing worn by the subject in the image:

- No Protective Clothing.
- With Head Protection.
- With Eye Protection.
- With Eye and Head Protection.

Disposable Respirators With Valve and Without Valve have an additional classification based on their degree of protection:

- FFP1.
- FFP3.
- FFP2 – N95 – KN95.
- Other – Unknown.

This classification is intended for future use and is achieved by searching for the model of the mask or adopting visual indicators. Some respirator manufacturers use different colors to distinguish the degree of protection. However, this classification does not apply to Full-Face and Half-Face Respirators, as their degree of protection depends on the filter used. It is also not feasible for Non-Medical and Surgical Masks, as they serve different purposes than respirators. This part of the database

TABLE 3.1: FMR-DB list of image labels. This table was taken from [68].

Label	**Meaning**
MS	With Facial Mask or Respirator
NM	Without Facial Mask or Respirator
WV	Disposable Respirators With Valve
NV	Disposable Respirators Without Valve
FF	Full-Face Respirators
HF	Half-Face Respirators
NM	Non-Medical Masks
SR	Surgical Masks
EP	With Eye Protection
HP	With Head Protection
EH	With Eye and Head Protection
OC	With Occlusions
NO	Without Occlusions
HM	Hands on Mouth
HT	Hats
NW	Neck Warmers and Bandanas
SN	Sunglasses

can help implement automatic systems for face detection and mask and respirator recognition.

3.3.1.4 Occlusions Classification

Images without facial masks or respirators are divided based on the presence or absence of occlusions. If occlusions are present, they are further classified by type. Eyeglasses and beards are not considered occlusions, as they are common features. The following are classified as occlusions:

- Hands on Mouth.
- Neck Warmer.
- Hat.
- Sunglasses.

The database contains 90 images for each type of occlusion and 315 images of people without occlusions. This part can help implement automatic face and occlusion detection systems.

3.3.2 Creation of a Mask Detection System with FMR-DB

The FMR-DB dataset was utilized to train an Artificial Neural Network (ANN) to recognize the type of mask or respirator worn. ANN have gained popularity recently due to their ability to solve complex problems with high precision, even without prior knowledge of the most essential features. Instead, they can deduce these features independently, albeit at the cost of information redundancy. In the following Sections, the creation of the mask detection system will be discussed in detail.

3.3.2.1 Artificial Neural Network Choice

The EfficientNetB0 [70] CNN architecture, depicted in Table 3.2, was chosen to address the problem of recognizing the type of mask or respirator worn. This state-of-the-art neural network is simple and has a relatively low number of parameters. It was implemented using Keras with Tensorflow as the backend [73, 74], and its model weights were initialized using the parameters previously learned by training the EfficientNet-B0 with the ImageNet database [75]. The top layers of the network

were replaced with a GlobalAveragePooling2D layer, a Dropout layer with a rate of 0.5, and a Dense layer activated by the softmax function to get the final prediction. The Adam optimizer compiled the network using the following parameters: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = e^{-6}$ [76]. An input image size of 128x128 pixels was set, resulting in the neural network requiring approximately 4 million parameters.

3.3.2.2 Hyperparameters Settings

The hyperparameters were set as follows for training the neural network:

- Batch size = 32.
- Learning Rate (LR) = 0.00025.
- Maximum number of epochs = 100.

A primary division of the dataset into $90\% - 10\%$ was performed to obtain the test dataset, followed by a 9-fold cross-validation on the first subset to obtain variable train and validation datasets and a final subdivision between datasets of $80\% - 10\% - 10\%$. Data augmentation was applied to the images with the respective parameters arranged as follows:

- Brightness range = (0.5, 1.0).
- Rotation range = $\pm 15°$.

TABLE 3.2: EfficientNet-B0 architecture. This table was taken from [68].

# Stage	**Operator** $\hat{F}_i$	**Resolution** $\hat{H}_i x \hat{W}_i$	**# Channels** $\hat{C}_i$	**# Layers** $\hat{L}_i$
1	Conv 3x3	224x224	32	1
2	MBConv1, k3x3	112x112	16	1
3	MBConv6, k3x3	112x112	24	2
4	MBConv6, k5x5	56x56	40	2
5	MBConv6, k3x3	28x28	80	3
6	MBConv6, k5x5	14x14	112	3
7	MBConv6, k5x5	14x14	192	4
8	MBConv6, k3x3	7x7	320	1
9	Conv 1x1 & Pooling & Fully Connected (FC)	7x7	1280	1

- Zoom range and width and height shift range = $\pm$ 0.05.
- Shear range = $\pm 2.5^{\circ}$.
- Zoom range = $\pm$ 0.05.
- Channel shift range = $\pm$ 50.
- Random horizontal flip enabled.

Some Keras training callbacks were used, such as EarlyStopping, which was set to stop training before the end of the 100^{th} epoch if the validation loss did not improve for ten consecutive epochs. Finally, a Z-Score normalization was performed on each image to improve the learning capacity of the neural network.

3.3.2.3 Training Configuration

The neural network was trained on a system with the following configuration:

- CPU: Intel Core i7 8750H
- GPU: NVIDIA GeForce RTX 2080 8GB GDDR6 with Max-Q Design
- RAM: 32 GB DDR4, running at 2666 MHz
- SSD: 2x2 TB PCIe M.2

TABLE 3.3: Training results. This table was taken from [68].

Fold	Test Accuracy	Epochs Of Training	Final Validation Accuracy	Final Validation Loss
0	0.98419	38	0.96109	0.11597
1	0.96443	23	0.97665	0.07127
2	***0.99209***	***40***	***0.95720***	***0.12897***
3	0.96838	18	0.93774	0.28144
4	0.96443	24	0.96498	0.12392
5	0.98024	24	0.97276	0.07326
6	0.97233	27	0.95331	0.20121
7	0.95257	20	0.96109	0.17073
8	0.97628	17	0.95703	0.12075

The GPU was chosen over the CPU for training the neural network due to its parallelization capabilities, significantly speeding up the training process. GPU-accelerated primitives were installed to utilize it, specifically the NVIDIA CUDA Toolkit [77] and the NVIDIA CUDA Deep Neural Network (cuDNN) library [78]. These libraries provide optimized implementations of common Deep Learning (DL) operations, allowing for efficient utilization of the processing power of the GPU.

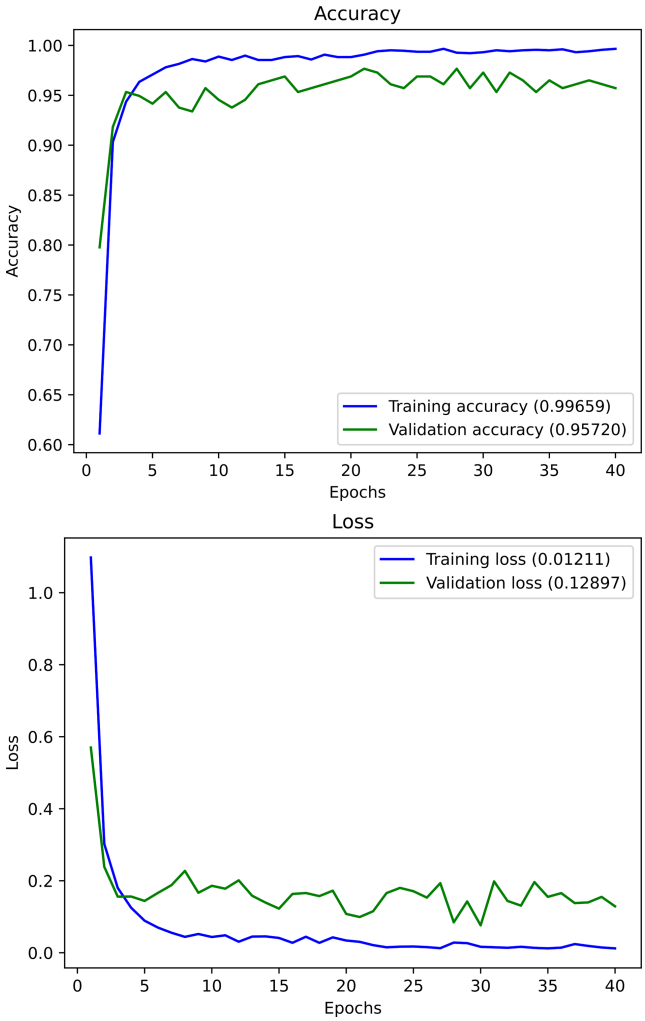

FIGURE 3.10: Accuracy and loss graphs related to training and validation of the neural network trained using Fold 2 as the validation dataset. This image was taken from [68].

3.3.2.4 Test with Some FMR-DB Images

The database can serve various purposes, but in this study, the authors have chosen to concentrate on identifying the type of mask or respirator worn by individuals. Nine distinct neural networks were trained using 9-fold cross-validation based on the settings outlined above. The outcomes of these training sessions are displayed in Table 3.3, with the best-performing neural network highlighted in gray. Overall, all

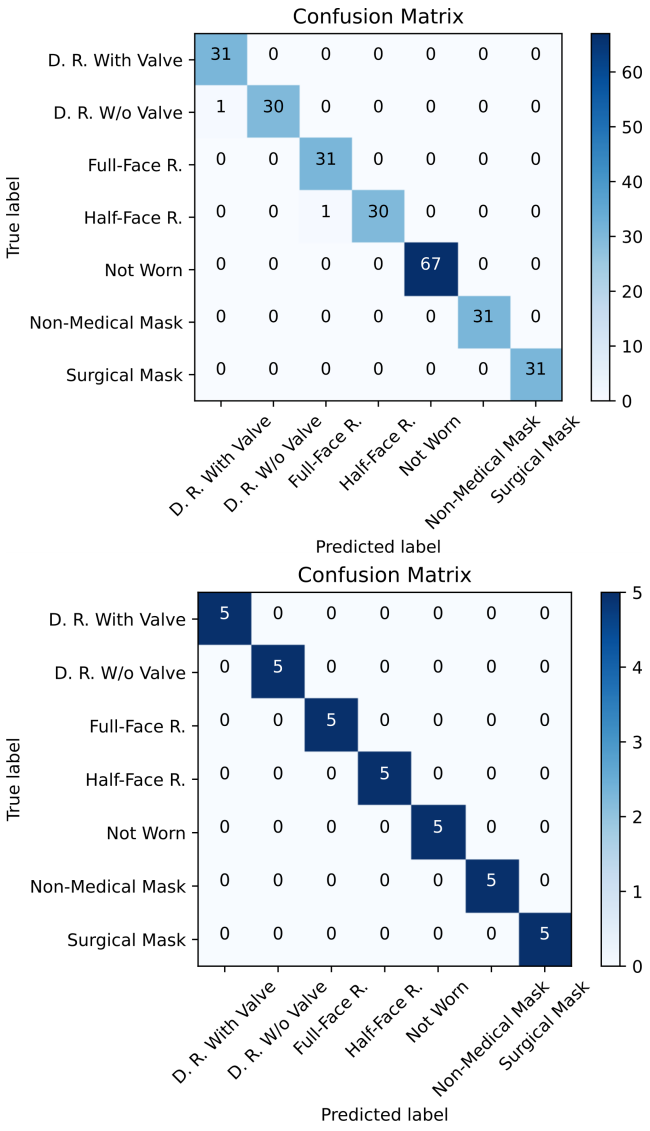

FIGURE 3.11: Comparison of confusion matrices of the neural network trained using Fold 2 as the validation dataset computed on the first test dataset (top) and the second (bottom). This image was taken from [68].

the trained neural networks yielded satisfactory results. Figure 3.10 demonstrates the progression of accuracy and loss during the training and validation phases of the top-performing network, which was obtained using Fold 2 as the validation dataset. This network achieved over 60% training accuracy and 80% validation accuracy within the first epoch alone. In subsequent epochs, these values improved rapidly, reaching a plateau. The image shown at the top of Figure 3.11 presents the confusion matrix calculated on the test dataset, where only two errors were made. A Disposable Respirator Without Valve was misidentified as one with valve, and a Half-Face Respirator was confused for a Full-Face one. These distinctions are pretty subtle. In the first case, the mistake was illustrated in Figure 3.9. Regarding the second case, if an individual is wearing a Half-Face Respirator in conjunction with an eye protection, it could be challenging, even for a human, to determine whether it is a Half-Face Respirator or a Full-Face one. This issue also arose during the classification of some database images.

3.3.2.5 Test with Other Images

Following the successful outcomes from the database test images, the system was evaluated further using additional images. Our research group has equipped five volunteers with the required gear and requested them to capture pictures of themselves donning various types of masks and respirators compatible with the database. Figure 3.9 shows a sample of these images. The results of this test are presented in the confusion matrix shown at the bottom of Figure 3.11, showcasing the robustness of the trained neural networks.

3.3.2.6 Comments on the Results Obtained

Through this project, our research group aimed to establish a valuable database readily accessible to the scientific community to develop such systems. While the current results are promising, there is potential for further enhancement of the FMR-DB. One approach is to expand the database size by incorporating new images, which would facilitate the creation of more robust automatic recognition systems than those currently attainable with the existing database version. Exploring different neural networks and parameters could also contribute to improving the results.

3.3.3 Ways to Wear a Mask or a Respirator Database

A further problem, often underestimated, consists in evaluating the correctness of the way in which masks or respirators are worn. If worn incorrectly, they entirely or partially cancel their protective effects and increase the spread of respiratory droplets and, therefore, the risk of transmitting viruses and bacteria. The second effort of the research group against the COVID-19 pandemic was then the creation of the Ways to Wear a Mask or a Respirator Database (WWMR-DB) [79, 80]. This database meticulously classifies how masks or respirators could be worn, making it suitable for evaluating mask detection systems and databases. The WWMR-DB was used to test a neural network, the ResNet-152 [81], which was trained on less detailed databases, the FMLD and the MaskedFace-Net [82, 83], to demonstrate their limitations in scenarios similar to real-world situations.

3.3.3.1 Database Creation

The database design was motivated by the following research question: *what are the most common ways people wear masks?* After a period of evaluation combining direct observation and related articles, eight representative classes were identified:

- Mask or Respirator Not Worn.
- Mask or Respirator Correctly Worn.
- Mask or Respirator Hanging from an Ear.
- Mask or Respirator Under the Nose.
- Mask Folded Above the Chin.
- Mask or Respirator Under the Chin.
- Mask or Respirator on the Tip of the Nose.
- Mask or Respirator on the Forehead.

A good system should work accurately regardless of face rotation, so $0°$, $45°$, and $90°$ were chosen to discretize the yaw rotations. An example of the previously mentioned classes and yaw rotations is illustrated in Figure 3.12. Due to COVID-19 restrictions

and safety concerns, taking pictures in a photographic laboratory was impossible. For this reason, a virtual approach using Google Forms was chosen so participants could submit selfies safely.
Volunteers could participate multiple times using four types of masks or respirators:

- Disposable Respirator With Valve.
- Disposable Respirator Without Valve.
- Surgical Mask.
- Non-Medical Mask.

To encourage participation, no constraints were placed on the number of photos, and participants could earn tickets for prize drawings based on their image submissions. Between September 2020 and January 2021, 42 participants submitted their images.

FIGURE 3.12: Example images of the classes present in the WWMR-DB with different jaw rotations. From top-left to right-bottom: Mask or Respirator Not Worn, Mask or Respirator Correctly Worn, Mask or Respirator Under the Nose, Mask or Respirator Under the Chin, Mask or Respirator Hanging from an Ear, Mask or Respirator on the Tip of the Nose, Mask Folded Above the Chin and Mask or Respirator on the Forehead. This image was taken from [79].

3.3.3.2 Database Characteristics

As partially outlined in the previous Section, the WWMR-DB comprises 1222 images featuring 42 individuals wearing masks or respirators with defined jaw rotations. The participant demographics are approximately 60% male and 40% female, with most individuals between 20 and 40 years old. Variations in image quality, position, and yaw rotation are present due to the lack of strict requirements. However, these variations increase image diversity, making the dataset more robust. Each image comes with two label files in PASCAL Visual Object Classes (VOC) [84] and You Only Look Once (YOLO) [85] formats, containing bounding box positioning information for the mask or respirator, the person's face, and the combination of the previous. The open-source program LabelImg was employed to create these labels [86]. The images are named using a format that indicates the subject number, position, yaw rotation, and mask or respirator type. This classification allows the WWMR-DB to be combined with other similarly classified databases, such as the FMR-DB and the Medical Mask dataset [87], to create systems that recognize mask or respirator types.

3.3.3.3 Other Information

For the *Mask Folded Above the Chin* class, participants were instructed to use only surgical or non-medical masks, as respirators are thicker and less likely to be worn in this manner. The WWMR-DB is publicly available in an open-access format. As Google Forms remains active, the database grows with new images. Anyone can contribute to the expansion of the dataset by submitting photos, which will be accurately classified, labeled, and made available to the research community [79].

3.3.4 Testing of Mask Detection System with WWMR-DB

This Section aims to test mask detection systems trained using different datasets with WWMR-DB, which offers detailed classification for mask and respirator positions. However, the following discussion uses only part of its potential because existing systems cannot detect mask use with such granularity. In this regard, *factor analysis*, one of the statistical techniques of multivariate analysis, can be used to simplify a set of complex variables or items by identifying the underlying dimensions or

factors that explain the relationships between them. As a result, the images were grouped into three superclasses: *Not Worn*, *Correctly Worn*, and *Incorrectly Worn*. The Correctly Worn superclass only includes images from the Mask or Respirator Correctly Worn class. The other classes belong to the Incorrectly Worn superclass, except for the Mask or Respirator Not Worn class. This strict configuration is referred to as *Rigorous*. Recognizing that the Rigorous configuration might be too stringent, a looser configuration called *Non-Rigorous* was defined. In this configuration, masks or respirators are considered Correctly Worn even if placed on the tip of the nose or exposed to the chin. However, these positions must still be considered incorrect, as they allow more air to pass through the gaps between the mask and nostrils or the fold created by the mask around the cheeks. Figure 3.13 shows the re-subdivision of WWMR-DB images for each class and configuration. Finally, the images of the WWMR-DB were cropped according to the bounding box labeled *person*, which identifies people's faces in the label files.

3.3.4.1 Evaluation of a Low-Granularity Dataset

The Face Mask Label Dataset (FMLD) has fewer classes than the WWMR-DB dataset and consists of images fetched from the Internet. An example is displayed in Figure 3.14. Initially, the FMLD was employed to train and test various neural networks to create an accurate mask detection pipeline. The ResNet-152 [81] achieved the highest performance, with a prediction accuracy of 98.79%. Since this ANN was published alongside the paper, it was chosen to conduct new tests using the same settings. The following is a summary of the settings assigned to it:

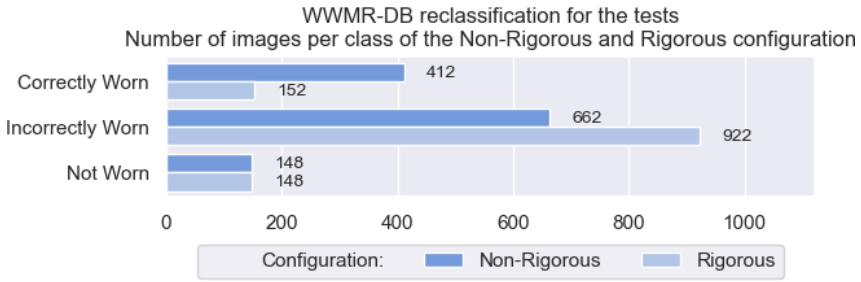

FIGURE 3.13: Reclassification of the WWMR-DB database images into the Rigorous and Non-Rigorous configurations. This image was taken from [79].

- The network outputs are the Compliant (masks worn correctly) and Non-Compliant (masks worn incorrectly or not worn) classes.
- The image size was set to 224x224 pixels, and a random 80:20 split of the training and validation set was performed.
- Transfer learning was applied by initializing the model weights with the parameters learned from pre-training the neural network with the ImageNet database [75].
- Cross-entropy was used as the learning objective.
- Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a learning rate of 0.001, reduced by a factor of 0.1 every seven epochs, was used as the optimization algorithm.
- The training was repeated 10 times for a maximum of 10 epochs, shuffling images and randomly dividing them between training and validation sets.

To adapt the Rigorous and Non-Rigorous configurations of the WWMR-DB to the Compliant - Non-Compliant classification and perform the test, the Incorrectly Worn and Not Worn superclasses were combined into the Non-Compliant class, while the Correctly Worn superclass was renamed as Compliant.

FIGURE 3.14: Some of the images originally present in the WIDER FACE and MAFA datasets and relabeled in the FMLD. This image was taken from [79].

3.3.4.2 Results with the WWMR-DB

In the Non-Rigorous configuration, the pre-trained neural network achieved an accuracy of 94.19%. The error analysis in Figure 3.15 shows that it is excellent at recognizing when a person is not wearing any mask or respirator. However, it tends to make more mistakes with faces rotated at 90° and masks or respirators worn under the nose. In the Rigorous configuration, accuracy decreased to 75.2%. This drop is

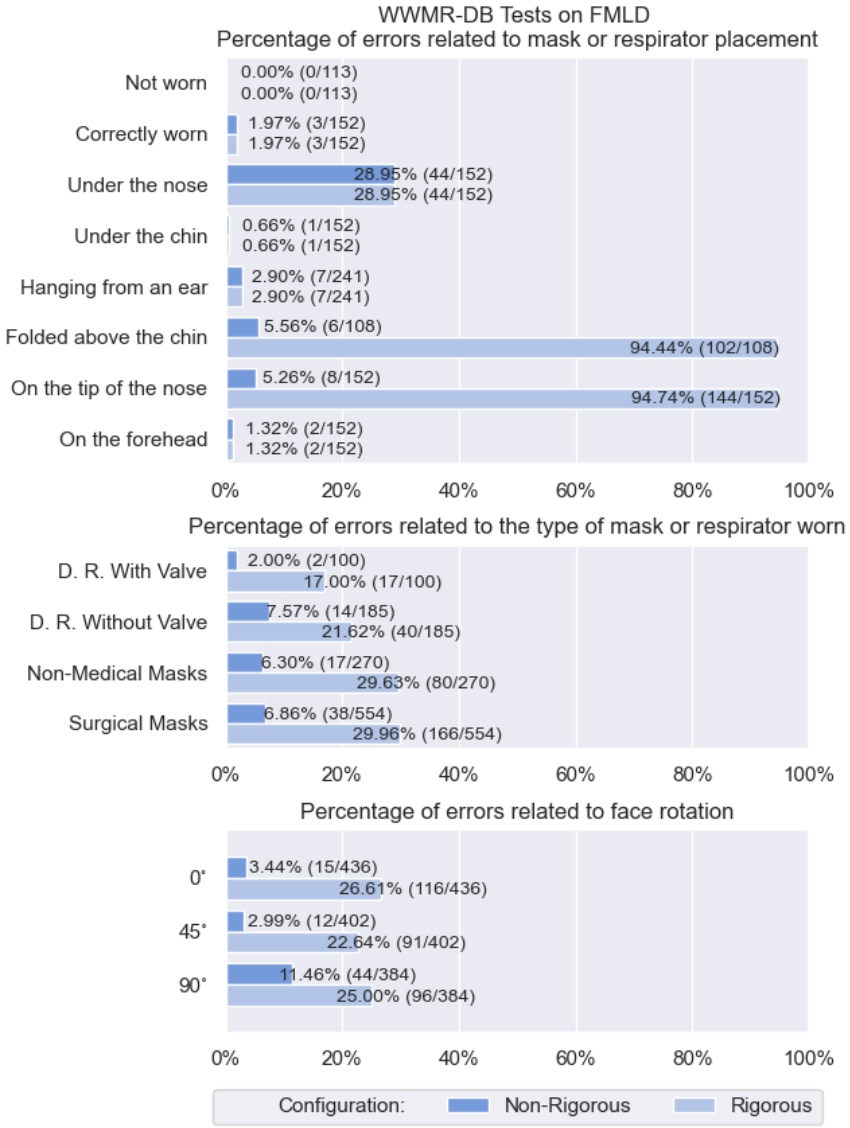

FIGURE 3.15: Percentage of errors due to the positioning and type of mask or respirator worn (top and center) and the rotation of the subject's face (bottom) during the WWMR-DB tests on FMLD. This image was taken from [79].

mainly because most images classified as Mask Folded Above the Chin or Mask or Respirator on the Tip of the Nose were predicted to be Correctly Worn. The confusion matrices in Figure 3.16 confirm that the pre-trained system performed relatively well in the first configuration but not in the second, which is more representative of real-world scenarios. In approximately 29% of cases, images classified as Under the Nose were predicted as Correctly Worn. In the Rigorous configuration, the two classes that differentiate it from the Non-Rigorous configuration had an error rate of around 95%. In conclusion, the neural network trained with FMLD performs adequately

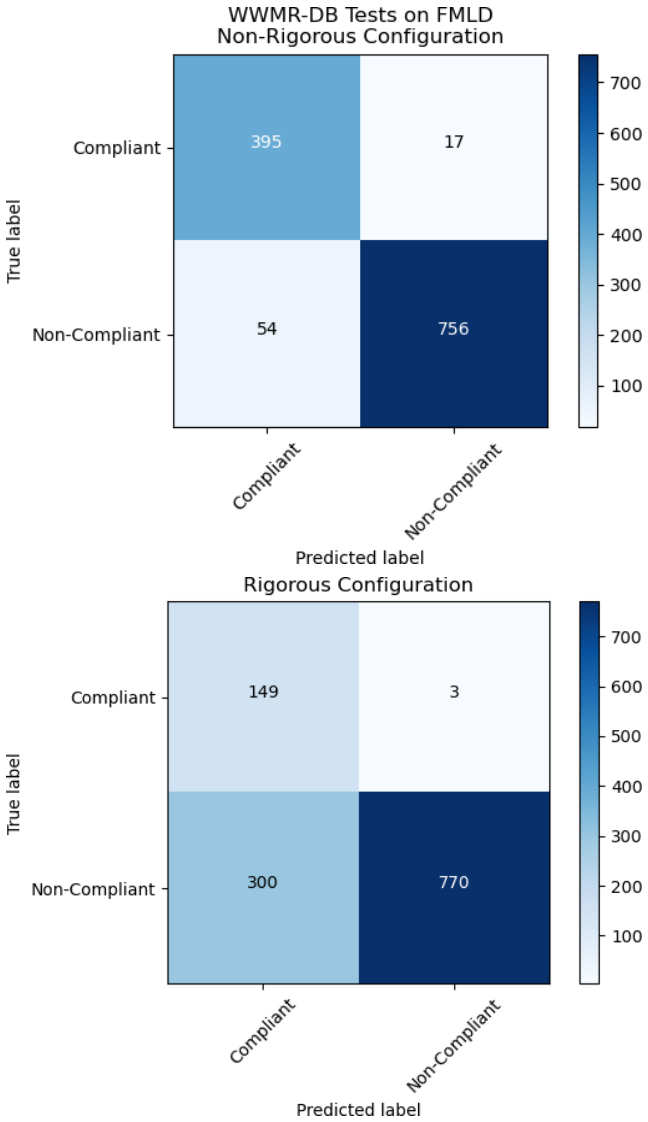

FIGURE 3.16: Comparison of confusion matrices of the tests performed with the neural network trained using FMLD dataset. This image was taken from [68].

in the Non-Rigorous configuration test. However, the Rigorous configuration test reveals that it cannot accurately model the entire mask detection problem.

3.3.4.3 Evaluation of a Dataset with Only One Mask Type

MaskedFace-Net [82] differs from the WWMR-DB as it consists of images from the Internet featuring only a blue surgical mask, the most common type available on the market. This mask was artificially superimposed on images from the Flickr-Faces-HQ3 (FFHQ) dataset [88], resulting in two datasets: the Correctly Masked Face Dataset (CMFD) and the Incorrectly Masked Face Dataset (IMFD). As the meanings of the acronyms themselves state, CMFD contains people wearing masks correctly, while IMFD contains people wearing masks incorrectly. Figure 3.17 shows an example of how an image from the FFHQ dataset was transformed to obtain corresponding images in the CMFD and IMFD. Since the Not Worn case is missing, the FFHQ dataset was used for it. CMFD, IMFD, and FFHQ datasets contain approximately 70000 images. Only a part of the dataset images was used for the training and validation phases to avoid increasing bias due to using the same images for training. Specifically, around 20000 images were used for each class, totaling approximately 60000. 80% of these images were used for training, and the remaining 20% for validation. During the testing phase, original and augmented versions of each unique image were used, totaling approximately 30000 images. Figure 3.18 provides a detailed image of the subdivision adopted. The same neural network (ResNet-152 [81]) and settings were used to enable better comparison with

FIGURE 3.17: Example of an image from the FFHQ dataset (left) and the images obtained by artificial transformations and contained in the CMFD (middle) and IMFD (right). This image was taken from [79].

previous results, except for the batch size, which was reduced to 16 due to memory constraints, and the application of data augmentation techniques to reduce overfitting.

3.3.4.4 Results

After training, the neural network was evaluated on the test dataset and achieved an impressive 99.96% accuracy. Only 16 misprediction errors were made, all related to the Incorrectly Worn class. The confusion matrix is not shown due to the exceptional results.

3.3.4.5 Results with the WWMR-DB

In the Non-Rigorous configuration, the neural network achieved an accuracy of 24.71%, while in the Rigorous configuration, the accuracy was 18.33%. These results may be due to insufficient database granularity and a need for more variety in mask types and colors. The error analysis and confusion matrices for both tests, shown in Figures 3.19 and 3.20, indicate that the neural network has only learned to recognize people not wearing masks. In other cases, there is a very high number of errors, even reaching 100% in the case of Mask or Respirator on the Forehead and Mask Folded Above the Chin, the latter case in the Rigorous configuration only. These results are comparable to those obtained using a random classifier, suggesting

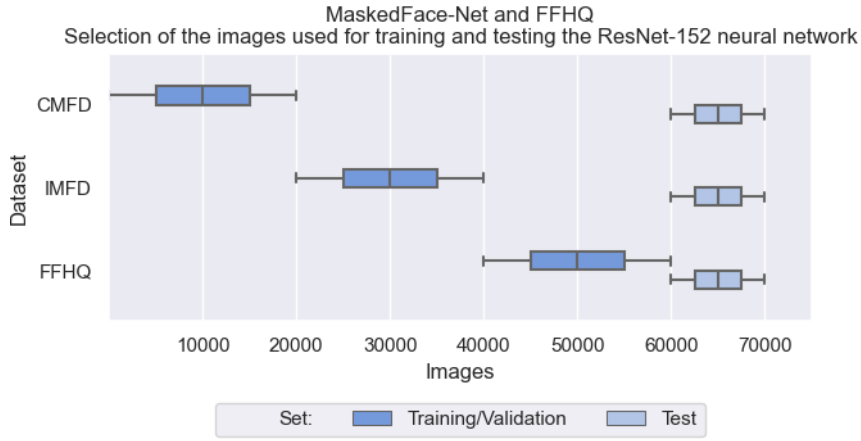

FIGURE 3.18: Subdivision of images used for training and testing the ResNet-152 [81] ANN. It allows for the reduction of bias due to the partial reuse of the same image. Care was taken to use different images for the training and validation phases, while this was not necessary for the testing phase. This image was taken from [79].

that MaskedFace-Net may not be effective for building systems that work well in real-world scenarios.

3.3.4.6 Comments on the Results Obtained

This research highlighted the impact of improper mask and respirator usage on the accuracy of detection systems. A new database called WWMR-DB was created to

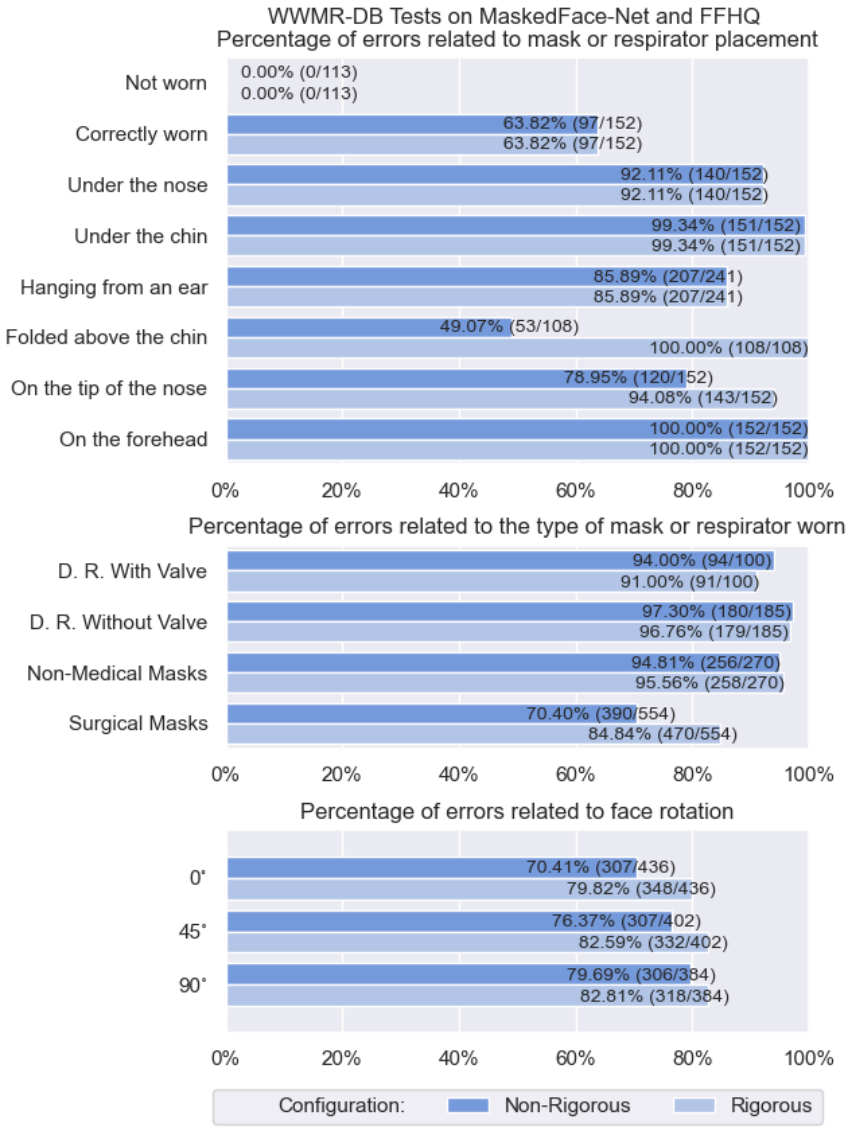

FIGURE 3.19: Percentage of errors due to the positioning and type of mask or respirator worn (top and center) and the rotation of the subject's face (bottom) during the WWMR-DB tests on MaskedFace-Net and FFHQ. This image was taken from [79].

evaluate these systems in situations similar to the real world. The results demonstrated that current mask and respirator detection systems are not fully equipped to handle this issue.

It is pointed out that it is necessary to consider the problem as at least ternary to solve it, so an intermediate class should be added to describe cases where the mask is not worn correctly. Moreover, the focus should be on how masks and respirators are worn, as this has been a significant cause of test failures. While some databases have already addressed the first point, the second issue must be satisfactorily re-

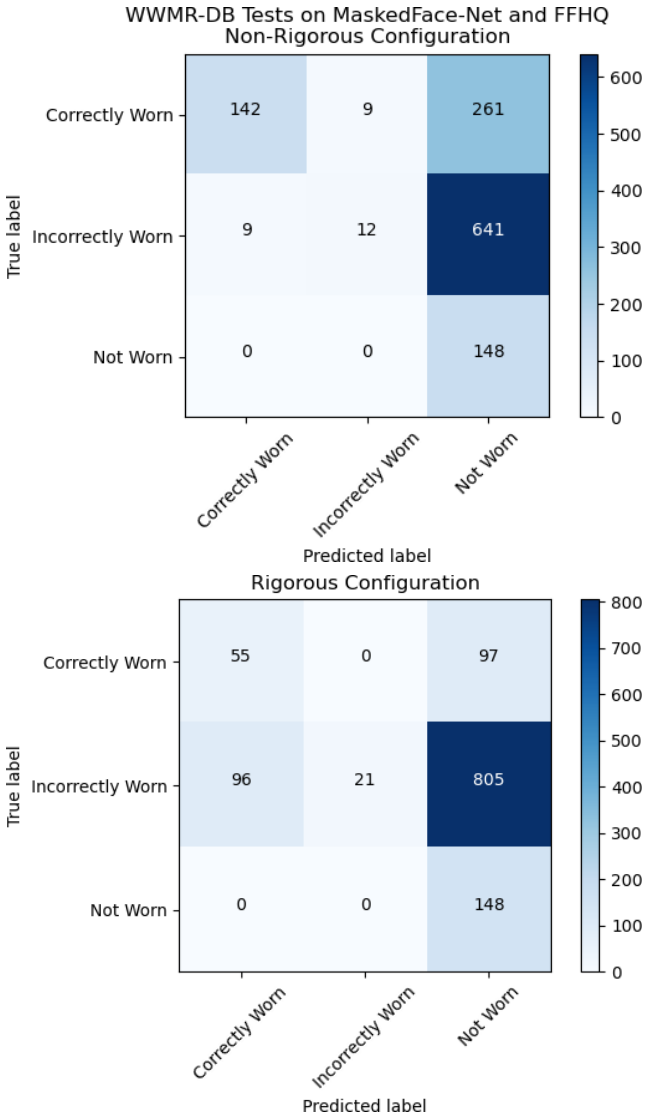

FIGURE 3.20: Comparison of confusion matrices of the tests performed with the neural network trained using MaskedFace-Net and FFHQ datasets. This image was taken from [68].

solved. Therefore, there is a need for richer databases with finer granularity, and the WWMR-DB classes can help fill this gap.

3.4 Waste Management

Based on the research conducted by the World Bank on waste management, it is estimated that nearly 4 billion tons of waste are produced globally each year, with urban waste making up a significant portion of that amount. In the near future, things do not seem to improve, as the accumulation of waste in developing countries is predicted to increase by 70% in the next 25 years [89]. As the number of industries in urban areas continues to grow, the waste management issue has become increasingly concerning. Waste typically consists of materials such as paper, plastic, metal, glass, and others. The primary method of waste management is landfills, which are costly, inefficient, pollute the natural environment, and pose health risks to nearby residents. Another common method is incineration, which causes air pollution and increases the likelihood of cancer [90]. It is then essential to recycle and repurpose waste in various ways so as to protect the environment.

Given that a significant portion of the waste generated in large cities is recyclable, it is crucial to understand and implement methods for reusing it to either provide benefits or at least reduce environmental issues. The availability of technologies or models to assist people in sorting waste is crucial for its proper disposal. Despite the existence of different recycling categories, many people still need to be made aware of how to determine the appropriate bin for each type of waste.

Effective waste management and classification are considered crucial for global ecological sustainability. Society must reduce waste accumulation through recycling and repurposing waste products. Proper sorting is typically used to enhance recycling and minimize environmental impact. In developing countries, waste management is a significant challenge in their urbanization and economic development and requires particular attention.

Numerous enterprises are actively participating in promoting waste management at various stages. Among them is ReLearn, an Italian startup that integrates technology and dynamic education with a data-driven methodology to inspire individuals to adopt eco-conscious behaviors regarding waste generation [25]. NANDO, ReLearn's AI system, elevates a standard waste bin into an intelligent bin by incorporating an

AI-powered sensor and an interactive screen.
In this Section, a collaborative project with ReLearn will be discussed [91]. It aims to develop a system capable of detecting and categorizing waste based on its type. To accomplish this, waste images provided by ReLearn were carefully labeled according to their material, which included glass, metal, paper, plastic, trash, and compost. These labeled images were then used to train a Mask Region-based Convolutional Neural Network (Mask-RCNN) [92] system for instance segmentation, whose framework is depicted in Figure 3.21, so as to verify whether the waste was correctly classified and recycled. Implementing such a system could enhance recycling efforts, increase knowledge and social engagement in waste classification and management, and improve community appeal by promoting sustainable practices.

3.4.1 Database Creation

The dataset is composed of 2451 images and includes six types of waste materials:

- Metal.
- Paper.
- Metal.
- Plastic.
- Glass.
- Trash.

The images used to create it were directly provided by ReLearn. The dataset was labeled in Common Objects in COntext (COCO) format using the VGG Image

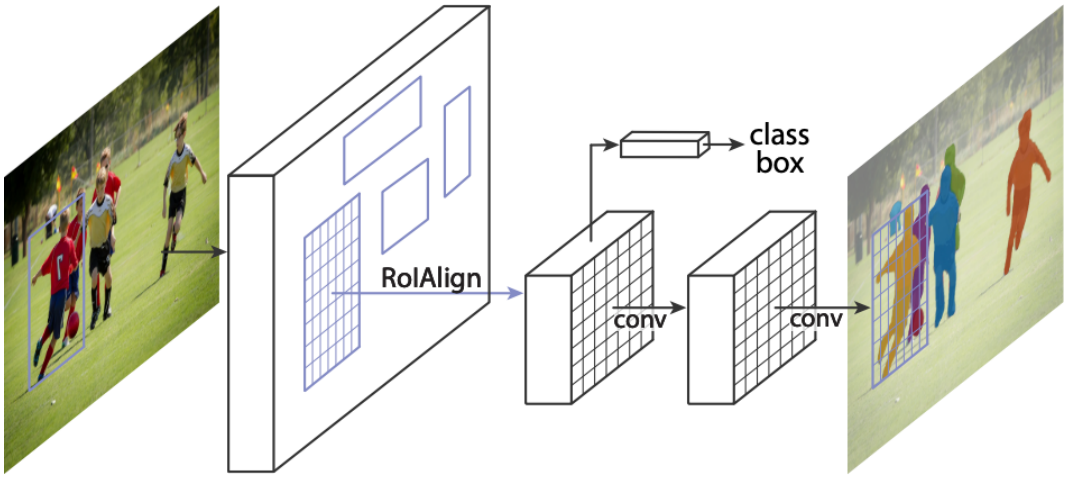

FIGURE 3.21: The Mask R-CNN framework for instance segmentation. This image was taken from [92].

Annotator (VIA) tool [93] [94], an annotation tool written in Python capable of annotating various objects and generating mask images. It can be run in a web browser without installation or configuration. A classification for the object type, such as plastic bottles, paper bags, tin cans, and others, was also added. Due to the Non-Disclosure Agreement (NDA) which was stipulated with the company, it is not possible to publish any photos of the dataset. Some examples from a publicly available dataset will be used as a graphic alternative [95].

3.4.2 Creation of the Waste Detection System

For this particular task, only the training and validation datasets were used. They were obtained by dividing the previously mentioned database by $80\% - 20\%$.

3.4.2.1 Hyperparameter Settings

The dataset was used to fine-tune a Mask-RCNN to obtain an instance segmentation system. Its hyperparameters were set as follows:

- Image size = 1024x1024.
- Batch size = 2.
- LR = 0.001.
- Maximum number of epochs = 100.

TABLE 3.4: AP and mAP of the trained system obtained on the training and the test dataset. This table was taken from [91].

Mask Shape	Dataset	Metal	Plastic	Paper	Glass	Trash	Compost	mAP
112x112	Training	0.8696	0.88	0.87	0.8696	0.8673	0.8685	0.8692
	Test	0.7811	0.7770	0.7817	0.7728	0.7597	0.7695	0.7736
56x56	Training	0.8797	***0.89***	***0.89***	0.8796	0.8770	***0.8797***	0.8793
	Test	0.8113	0.7866	0.8135	0.8153	0.8058	**0.7939**	0.8094
28x28	Training	***0.8851***	0.8817	0.8879	***0.8831***	0.8715	0.8732	***0.8804***
	Test	**0.8139**	**0.8671**	**0.8408**	**0.8554**	**0.8204**	0.7737	**0.8285**
11x11	Training	0.8776	0.8851	0.8857	0.8799	***0.8773***	0.8695	0.8791
	Test	0.8083	0.8257	0.8187	0.8075	0.7858	0.7934	0.8065

- Maximum number of steps per epoch = 1000.

Mask shape is another crucial parameter for Mask-RCNN. During the training phase, it extracts local features corresponding to different categories. In contrast, during the prediction process, it generates the shape of the object mask. Therefore, mask shape was tested at 112x112, 56x56, 28x28, and 11x11 in order to improve performance.

3.4.2.2 Training Configuration

The instance segmentation system was trained on a system with the following configuration:

- CPU: Intel Xeon E5-2683 v3.
- GPU: NVIDIA GeForce RTX 2080 Ti 11GB GDDR6.

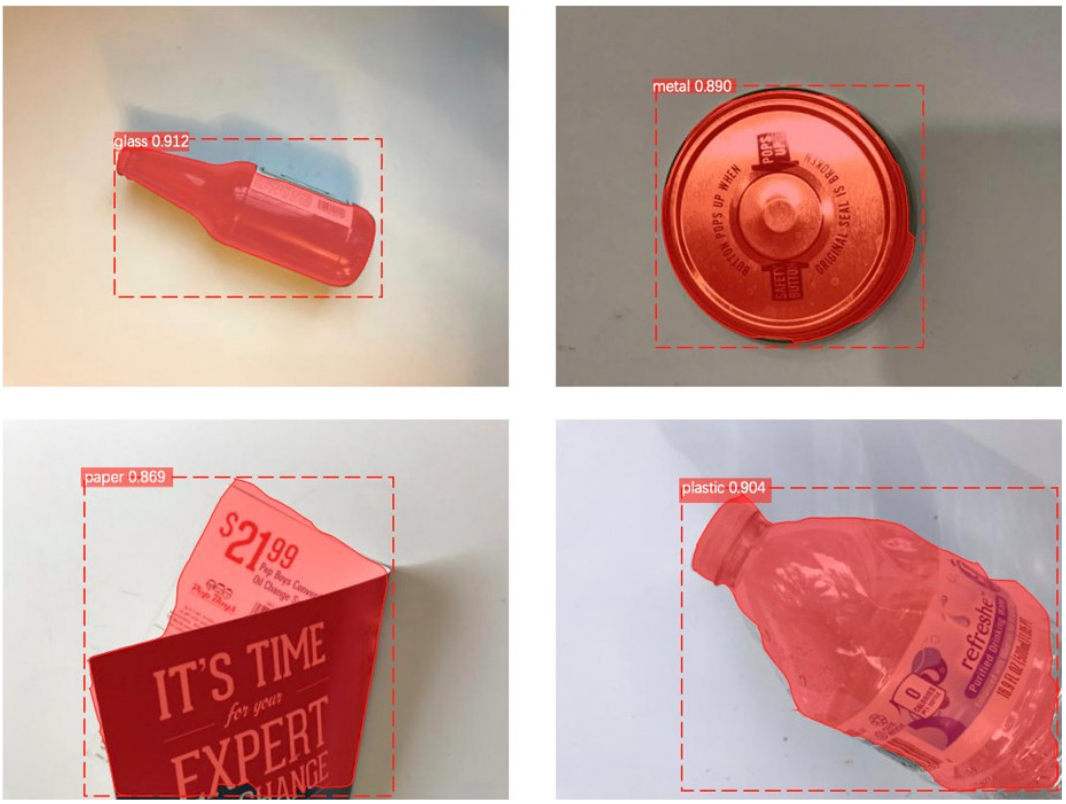

FIGURE 3.22: Some prediction results obtained using the publicly available dataset [95]. This image was taken from [91].

- RAM: 32 GB DDR4.
- SSD: 2 TB PCIe M.2.

As in the previous cases, the GPU was chosen over the CPU for training the system. This was also possible by recurring to the NVIDIA CUDA Toolkit [77] and the NVIDIA CUDA Deep Neural Network (cuDNN) library [78].

3.4.3 Training Results

Table 3.4 shows the test results performed after the training of different waste detection systems employing different mask shapes. In general, the results were quite satisfactory. However, it can be seen that, when looking at the Average Precisions (APs) relative to the different classes and the Mean Average Precisions (mAPs) relative to the different mask shapes, it is immediately verifiable that the best results were obtained using a mask shape of 28x28. The effectiveness of the best system was also tested on the publicly available dataset, obtaining good results. Figure 3.22 shows an example of this.

Chapter 4

Photoplethysmography Sensors

This chapter discusses an innovative Internet of Things (IoT) system designed to enhance *affective learning* by employing redundant non-invasive sensors, such as camera and Photoplethysmography (PPG) ones, to gather real-time feedback on students' *attention levels* based on their reactions. The primary goal of this system is to improve the learning experience and promote engagement by adaptively responding to students' emotional and cognitive states.

4.1 Photoplethysmography

PPG is a non-invasive optical technique used to measure changes in the blood volume of a vascular tissue bed [96, 97]. It has become increasingly popular in clinical and consumer devices due to its non-invasive nature and cost-effectiveness [98]. However, signal processing research has grown significantly with the widespread adoption of PPG sensors in wearable devices. This new context presents several challenges, such as handling motion artifacts, and the PPG signal contains valuable information on respiratory, cardiovascular, and Autonomic Nervous System (ANS) that still needs to be fully utilized. Combining these factors offers the potential to use PPG signals to provide detailed health information unobtrusively in daily life. To achieve this, developing robust PPG signal processing algorithms is crucial [99].

4.1.1 Physiological Background

PPG involves illuminating peripheral tissue with optical radiation, which is then scattered and absorbed as it passes through different tissue layers. The attenuated light intensity is detected by an optical sensor and recorded as a voltage signal known as the PPG waveform. As shown in Figure 4.1, raw PPG waveform reflects variations in the attenuation of incident optical radiation by different tissue components within the tissue volume [100]. The Alternating Current (AC) part contains high-frequency variations due to changes in arterial blood volume with each heartbeat. In contrast, the Direct Current (DC) part contains lower-frequency variations due to changes in other tissue components such as venous and capillary blood, bloodless tissue, and more. The attenuation of light in tissue can be described as a function of the optical path length and the attenuation coefficient of the medium, based on the modified Beer-Lambert law [101]. The origins of the PPG waveform have been attributed to red blood cell orientation, the mechanical movement of cellular components, and a combination of factors [102–104, 99].

4.1.2 Photoplethysmography Signal

The PPG signal has a quasi-periodic pattern consisting of an arterial pulse wave for each heartbeat. The pulse wave resembles an arterial blood pressure pulse wave but has distinct differences in waveform contour [107, 108]. As shown in Figure 4.2, PPG pulse wave has two phases: anacrotic (rising limb) and catacrotic (falling limb)

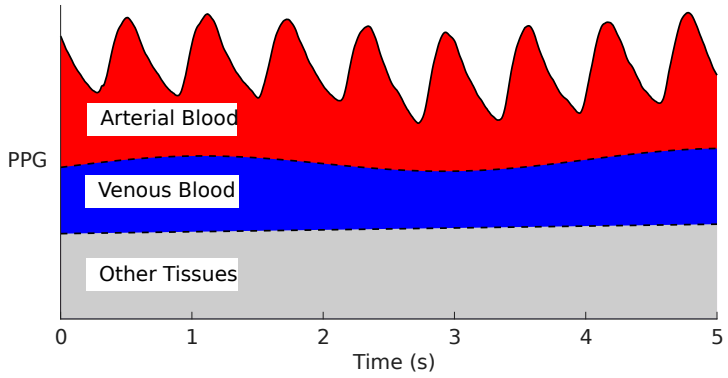

FIGURE 4.1: Changes in venous and arterial blood volume and other tissues affect the PPG signal. This image was taken from [105].

[108]. The anacrotic phase is dominated by systolic ejection, which is caused by the expansion of the arterial system due to the inflow of blood [98]. The amplitude of the systolic peak is related to the stroke volume, while the rate of expansion is related to the contractility of the heart. On the other hand, the catacrotic phase is dominated by wave reflections from the periphery [98]. The distance between two systolic peaks is defined as the Peak-to-Peak (PP) interval. The dichrotic notch and diastolic peak in the PPG waveform are caused by wave reflection, and their timing and location are influenced by arterial stiffness [107]. Diastolic decay is determined by the exponential contraction of the arterial system due to blood outflow and is influenced by resistance and vascular compliance. The diastolic peak is clearly visible in young individuals and tends to decrease with age [109]. In summary, the morphology of the PPG pulse wave is influenced by various factors [98, 108, 110, 111]:

- Circulation: arterial stiffness and Blood Pressure (BP).
- Heart: Heart Rate (HR), rhythm, and stroke volume.
- Other physiological processes: respiration and the ANS, which can be impacted by stress.
- Disease.

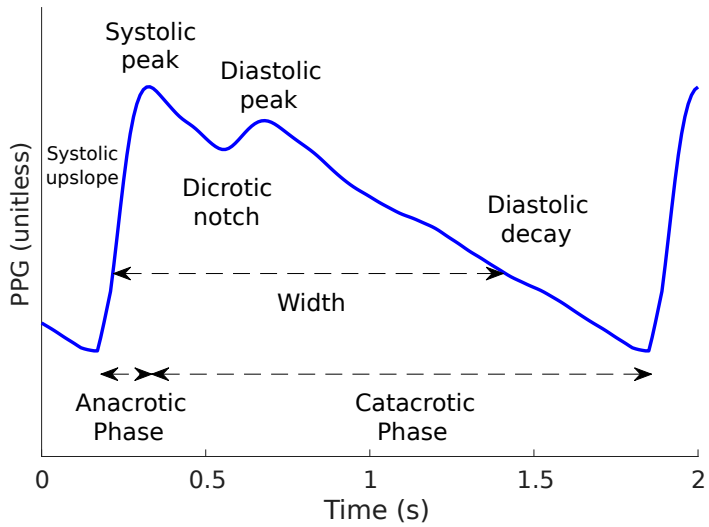

FIGURE 4.2: The morphology of the pulse wave PPG is influenced by circulation, heart, other physiological processes, and disease. This image was taken from [106].

PPG signal analysis faces several challenges, making extracting reliable information complex. The PPG signal exhibits various physiological variations; only one or some of these may be relevant to a particular analysis. It is also susceptible to different types of noise, such as powerline interference, probe-tissue interface disturbance, motion artifacts, and Low Frequency (LF) and High Frequency (HF) noise [98, 108]. Technical aspects like the measurement site location and the typology of the sensor used affect the waveform [108, 112]. The wide range of influences on the PPG signal adds complexity to its analysis [99].

4.1.3 Physiological Parameters

This Section will discuss HR and Heart Rate Variability (HRV), which are essential physiological parameters often analyzed in clinical settings [113]. HR is the number of heartbeats per minute, while HRV refers to the fluctuations in the time intervals between adjacent heartbeats [114]. HRV is an emergent property of interdependent regulatory systems, such as heart-brain interactions and dynamic non-linear ANS processes. These operate on different time scales to help humans adapt to environmental and psychological challenges. A healthy heart does not beat like a metronome; its Beat-to-Beat (BB) fluctuations are complex and non-linear, best described by mathematical chaos [115]. HRV regulates autonomic balance, BP, gas exchange, gut, heart, vascular tone, and possibly facial muscles [116].
Healthy oscillations of the heart provide the flexibility to cope with an uncertain and rapidly changing environment [117]. While higher HRV is generally associated with health and self-regulatory capacity, adaptability, or resilience, it is not always better. Pathological conditions can produce elevated HRV measurements, which are strongly linked to increased mortality risk, particularly among older people. Close examination of Electrocardiogram (ECG) morphology can reveal whether elevated HRV values are due to problems like atrial fibrillation [118].
Higher levels of resting vagally-mediated HRV are linked to the performance of executive functions like attention and emotional processing by the prefrontal cortex [114]. Afferent information processing by the Intrinsic Cardiac Nervous System (ICNS) can modulate frontocortical activity and impact higher-level functions [119]. In summary, HRV is an essential indicator of cardiac health, adaptability, and self-regulatory capacity, with implications for cognitive and emotional processing.

4.1.3.1 A Short Overview of HRV Metrics

Time-domain, frequency-domain, and non-linear measurements can be used to describe HRV for 24-hour, Short-Term (ST, approximately 5 minutes), or Ultra-Short-Term (UST, less than 5 minutes) periods. In particular:

- Time-domain measurements quantify the variability in measurements of the Interbeat Interval (IBI), which is the period between successive heartbeats. These values may be expressed in original units or as the natural logarithm (ln) of original units to achieve a more normal distribution [120]. Some of the most important time-domain measurements are the following [113]:
  - NN50: Number of Pairs of Adjacent NN Intervals Differing by More Than 50 ms.
  - pNN50: Proportion of NN50 Divided by Total Number of NNs.
  - SDNN: Standard Deviation of all Normal to Normal Intervals.
  - SDSD: Standard Deviation of Differences Between Adjacent Intervals.
  - RMSSD: Root Mean Square of Differences Between Adjacent Intervals.

  where NN stands for Normal to Normal and is tightly correlated with the PP interval since both represent a completed cardiac cycle.
- Frequency-domain measurements estimate the distribution of absolute or relative power into four frequency bands: High Frequency (HF), Low Frequency (LF), Very-Low Frequency (VLF), and Ultra-Low Frequency (ULF) bands. Power is the signal energy within a frequency band and can be expressed in absolute or relative power. The most important frequency-domain measurements concern the power and peaks of the different bands.
- Non-linear measurements allow to quantify the unpredictability of a time series, providing additional information about the complex dynamics of HRV [121]. Some of the most important non-linear measurements are the following [113]:
  - S: Area of the Ellipse Representing Total Heart Rate Variability.
  - SD1: Standard Deviation Perpendicular the Poincaré Plot Identity Line.
  - SD2: Standard Deviation Along the Poincaré Plot Identity Line.

4.2 Improving Student Engagement During Class

Teaching requires continuous self-reflection and evaluating the impact of the explanations provided. This skill usually develops with experience as teachers see how students respond to different teaching methods and content. However, this process can be more straightforward in smaller classrooms. In larger classrooms, especially those with fifty or more students, it becomes more challenging to interact with students and ensure everyone can see and hear the lessons. For this reason, a new IoT system based on non-invasive techniques was developed to help teachers receive feedback on students' attention levels [122]. It relies on facial expressions and physiological data acquired through common cameras and smartwatches. The idea is to use facial expressions as the primary data source, which can be validated using physiological data analysis, potentially leading to the development of a system that relies solely on facial expressions. Such a system would offer several advantages:

- It would be less invasive, as it would not require wearable devices to collect physiological data.
- It would be applied to in-person and distance learning lessons, making it versatile for various learning environments.
- It would be cost-effective, as it only requires webcams or similar devices for data collection.

These benefits are significant, as previous systems have relied on multimodal acquisition methodologies that are not easily implementable in everyday use. For instance, compared to the study in [123], this project uses consumer-grade devices, resulting in lower costs. Advances in technology have allowed for similar results using fewer parameters, as [123] recurs to Blood Volume Pulse (BVP), Skin Conductance (SC), and HR, while this project only recurs to HR and HRV. In [124], tests were conducted with a similar number of students but in a context regarding a writing activity. The face was traced using a Microsoft Kinect, while HR and HRV were extracted using ECG. However, these technologies are not readily applicable in realistic environments. Furthermore, engagement was estimated using Local Binary Patterns in Three Orthogonal Planes (LBP-TOP) and a machine learning tool with custom classifiers. In contrast, this project focuses on developing a more practical and adaptable system for attention detection in learning environments.

4.2.1 Background

Human cognitive abilities have been extensively explored from neurophysiological and psychological perspectives. This research has led to the development of various theories and educational methodologies aimed at optimizing the cognitive processes of the human mind [125]. As computers and electronic devices become increasingly prevalent in in-person and distance learning environments, the potential for distraction may negatively impact students' attention levels [126, 127].
Research indicates a strong connection between various cognitive processes and physiological responses, which can manifest in phenomena such as HR and eye blinking. By observing these responses, one can determine the cognitive state of an individual [128]. These factors have been collected and analyzed to identify attention and emotional elements that may influence learning.
In [129], previous assumptions suggest that students' attention wanes over time are disproven through physiological measurements, presented as a more objective method for assessing engagement levels. A comprehensive review of the neurophysiological variables used to determine attention levels can be found in [130]. These variables can be categorized into the Central Nervous System (CNS) and the ANS. Measurements of the former group typically require invasive techniques, which can limit student comfort and the large-scale implementation of such systems. Examples of this approach can be seen in [131, 132], where authors utilize Electroencephalography (EEG) to assess attention and learning levels. In contrast, the latter group pertains to signals that are commonly represented and modeled using facial expressions, eye-based measurements, HR, BP, and SC, offering less intrusive measurement methods for students. Detecting attention and emotional factors is closely associated with ANS signals and can be divided into two primary approaches: facial expressions and multi-neurophysiological signal detection.

4.2.2 Internet of Things for Affective Learning

The proliferation of IoT devices has resulted in billions of connected devices performing various tasks, ranging from personal use to larger company or societal purposes, such as fitness tracking or environmental monitoring [133]. These devices generate valuable information that can be analyzed for decision-making or lifestyle adaptations. However, the vast landscape of sensing and actuating devices presents

challenges regarding scalability and system integration [134]. The Social Internet of Things (SIoT) and related regulations have been proposed to address these issues [135–137].

The application of IoT technologies in the educational field has become a support technology and a research trend, aiming to improve traditional classrooms by collecting and analyzing data from educational contexts [138]. The aim of the work that will be presented in the following Sections is to leverage the potential of the IoT domain to enhance feedback collection for teachers seeking to engage students more effectively. Other researchers have already explored the merging of IoT and education. In [139], the authors used a WiFi-enabled Radio Frequency Identification (RFID) reader to assist teachers in automatically recording attendance and student behavior. They combined this with Quick Response (QR) codes for quick access to course materials and real-time interactive responses to stimulate participation. Implementing this system has reportedly increased participation rates from 85% to 98%. In 2021, [140] presented a voice assistant application for Google Home that interacts with teachers and students based on textbook contexts. Their model employs machine learning algorithms to recognize user sentences and learn new ones, with two case studies conducted in Bangladesh. In the realm of affective learning, [141] proposed an IoT-based framework to detect human emotions, specifically for special education. Wearable devices were used to gather data on HR, movement, and environmental factors like temperature and humidity. This information was then used to monitor students with attention deficit hyperactivity disorder and provide affective feedback to educators.

An alternative IoT framework was proposed in [142], where the authors developed an emotion recognition system analyzing diverse datasets from ANS and CNS signals. They conducted a transmission analysis to ensure real-time communication for emotion identified with an IoT hub. The authors aimed to provide a solution for distance learning and healthcare support during the Coronavirus Disease 2019 (COVID-19) pandemic.

4.2.3 Facial Expressions

Research on emotion recognition from facial expressions has been ongoing for many years. In 1971, American psychologists Paul Ekman and William Friesen published a list of six basic emotions shared by all human groups, regardless of culture: *anger*,

disgust, *fear*, *happiness*, *sadness*, and *surprise* [143]. These emotions were further expanded in subsequent works by Ekman, Daniel Cordaro, and other researchers [144, 145]. In 1978, Ekman introduced the Facial Action Coding System (FACS), which classifies the muscular movements of the human face [146]. FACS identifies 46 Action Unit (AU) responsible for facial movements. Combining multiple actions and their respective intensities generates many possible facial expressions. Only experienced annotators can accurately classify facial expressions using FACS. In [147], the authors discuss the channels of information used to convey emotions, recognizing facial expressions and vocal intonations as the primary ones. They divide the process of affect recognition from facial expressions into three stages: face detection, facial features extraction, and the description of the affective state. The first two stages have clear objectives, while the latter is more complex due to the subjectivity of the task. The authors suggest that the description of the emotional state should align with a human's description of the same affective state. However, they acknowledge that this interpretation can vary between individuals and cultures, and thus, affect recognition programs should be tailored to the context in which they are used.

Facial feature extraction is often performed by recognizing AUs of FACS rather than prototypic expressions of common emotions (happiness, anger, etc.). This is because prototypic expressions occur infrequently, while emotions are often conveyed by changes in discrete facial features [148]. Additionally, AU are studied by psychologists for the recognition of complex emotional states [149]. The study presented in [150] aimed to improve some shortcomings of FACS, such as the lack of temporal and detailed spatial information, by introducing a new representation called FACS+. Similar techniques, based on optical flow, are also used in [151] to develop a time-aware method for facial expression recognition. The survey in [149] analyzes standard techniques for affect recognition by breaking down their pipeline into fundamental components: face registration, representation, dimensionality reduction, and recognition. The face can be initially registered as a whole, a combination of parts, or by the localization of fiducial points. It is then encoded in a spatial or spatiotemporal representation. Spatial representation encodes image sequences frame by frame and can be performed at different levels of abstraction.

Datasets used to train and validate affect recognition techniques vary in labeling, expressions, and sources. Some use FACS for labeling, while others report prototypical expressions. Some datasets contain posed expressions, sometimes performed

by actors, while others are generated from labeling natural images. They can also contain subtle or exaggerated expressions. Pictures can be explicitly created for emotion recognition or taken from the Internet and movies.

4.2.3.1 Facial Expression Recognition

With the rise in popularity of neural networks, they have been increasingly used to improve affect recognition tasks. In [152], the authors combine a standard pipeline with traditional Gabor Filters and Local Binary Patternss (LBPs) along with Extreme Learning Machine (ELM) for real-time applications. Deep Learning (DL) models are also used but face limitations due to the limited number of available images [153]. To address this issue, a Shallow Convolutional Neural Network (CNN) is used in [154] to recognize micro-expressions with limited training samples. In [153], domain knowledge guides the model toward relevant facial areas. In [155], a Deep Belief Network (DBN) is trained to perform feature learning, feature selection, and classifier construction in a unified loopy framework. An identity-aware CNN is introduced in [156] to mitigate variations caused by personal attributes, resulting in improved facial expression recognition performance. A Contractive Convolutional Network (CCNET) is used in [157] to achieve invariance to translations of facial traits in the image.

4.2.3.2 Training a Neural Network for Facial Expression Recognition

In recent years, numerous high-quality facial expression databases have been released. In [158–162], the CNN presented in [153] was trained using the Keras library [73] by combining different facial expression databases, called Database Ensembles (DEs). This strengthens the trained system as they:

1. Increases the size of the train, validation, and test datasets.
2. Have diverse backgrounds.
3. Have different photographic quality.
4. Portrays various human traits.

This procedure generally allows the creation of Artificial Intelligence (AI) systems that can generalize better and remain reliable even under conditions that differ from those used during training. Since facial expressions can be either posed or spontaneous, two DEs were created:

1. Ensemble 1, containing images from databases with both posed and spontaneous facial expressions:
   (a) Extended Cohn-Kanade Dataset (CK+) [163] [164].
   (b) FACES Database [165].
   (c) Facial Expression Recognition 2013 (FER2013) [166] plus FER2013+ annotations [167].
   (d) Japanese Female Facial Expression (JAFFE) [168].
   (e) Multimedia Understanding Group (MUG) [169].
   (f) Radboud Faces Database (RaFD) [170].
   (g) Static Facial Expressions in the Wild 2.0 (SFEW 2.0) [171].
2. Ensemble 2, containing images from databases with only posed facial expressions:
   (a) CK+ [163] [164].
   (b) FACES Database [165].
   (c) JAFFE [168].
   (d) MUG [169].
   (e) RaFD [170].

For reasons explained later, the authors used the neural network obtained from Ensemble 1, which consisted of seven different databases with a total of 43993 photos, divided into eight different facial expressions representing emotions: *anger*, *contempt*, *disgust*, *fear*, *happiness*, *neutrality*, *sadness*, and *surprise*. The creation of this DE was facilitated by using the Facial Expressions Databases Classifier (FEDC) [160]. The images were preprocessed by applying the following transformations:

- Face cropping.

- Grayscale conversion.
- Resize to 48x48 pixels.
- Z-score normalization.

A $90\% - 10\%$ random split was performed to obtain training and test datasets, while the validation dataset was obtained via 9-fold cross-validation on the first subset, resulting in a final split of $80\% - 10\% - 10\%$. During each training epoch, the following data augmentation operations were applied:

- Brightness range between 50% and 100%.
- Random horizontal flip.
- Random rotation among $\pm 2.5^{\circ}$.
- Shear range among $\pm 2.5\%$.
- Width and height shift range among $\pm 2.5\%$.
- Zoom transformation interval among $\pm 2.5\%$.

The best fold combination for the CNN described in [153], using Ensemble 1 as the train, validation, and test dataset, achieved a test accuracy of 80.38%. The respective confusion matrix is displayed in Figure 4.3. This accuracy is lower than the accuracies obtained using only CK+ (92.5%) and Ensemble 2 (96.78%) datasets. However, such datasets present some problems:

- They comprise images taken in a photographic studio, implying less variability than real-world use cases.
- They only represent posed facial expressions, creating a bias towards more frequently used spontaneous expressions, which are more natural but have less inter-class variability.
- They contain few images, reducing the ability of the neural network to generalize and perform well in real-world scenarios.

Due to these reasons, despite having worse quantitative metrics, the model obtained using Ensemble 1 as the train, validation, and test dataset was considered reliable for the facial expression recognition problem and employed for the proposed approach.

4.2.4 Attention Detection Algorithm

This Section will discuss the implementation of the attention detection algorithm for the two data sources of interest in this work: facial expressions and physiological data.

4.2.4.1 Attention Detection Algorithm Based on Facial Expressions

Facial expression recognition was accomplished using the Emotion Detector (ED) software [158–162], which was developed using Java, Apache Maven, OpenCV [172], and Deeplearning4j (Dl4j) [173]. It performs facial expression recognition using the neural network discussed in Section 4.2.3.2 and saves the result as individual images or a single Comma-Separated Values (CSV) file. Videos from in-person and distance learning lessons were analyzed, and two samples were captured per second. The results were exported to CSV format for further analysis.
To align facial expression predictions obtained through CNN with physiological data, they were grouped into active and passive moods based on a Russell-like classification [174]. Active moods correspond to high-arousal emotions (anger, fear, happiness, surprise), while passive moods correspond to low-arousal emotions (contempt, disgust, sadness). Facial expressions depicting *neutrality* were excluded from

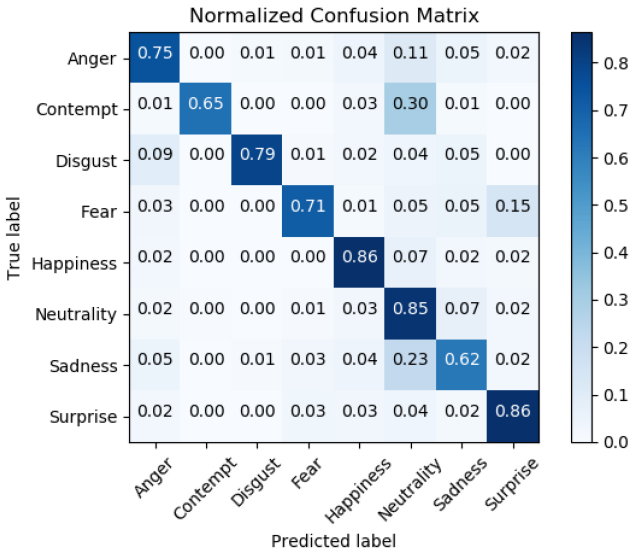

FIGURE 4.3: Normalized confusion matrix of the neural network discussed in [153], trained recurring to Ensemble 1 dataset. This image was taken from [158].

this classification.
An activation level was determined by counting active mood readouts within 100 seconds and normalizing it to $Ws_{NN} = 300$. A binary classification was applied to the activation level by comparing it with a threshold T_{fe} corresponding to $\frac{Ws_{NN}}{3}$. This resulted in two different levels stored in an AA from facial expressions (AA_{fe}). The binary array contains a value for each time interval, labeling it as *Attention* if the attention grade is higher than T_{fe}, or *Distraction* otherwise.

4.2.4.2 Attention Detection Algorithm Based on Physiological Reaction

The attention detection algorithm based on physiological reactions was built using data acquired through a set of Venu Sq, which are commercial smartwatches made by Garmin. Physiological data, including HR and HRV, were acquired with a sampling rate of $1Hz$. The read values were gradually transferred via Bluetooth Low Energy (BLE) to a series of smartphones to be saved in a text file.
The algorithm itself was developed using MATLAB and executed after a two-level calibration phase focused on estimating the threshold for emotional phase detection. This calibration was performed for each individual based on the information (age, sex, weight, and height) and the estimate of the initial condition.
The resulting algorithm can be run live during a lesson to perform behavioral analysis. It works as follows:

1. A certain number of N samples are observed at a time, related to HR and HRV.
2. From this observation window, defined as Window Size (WS), SDNN, RMSSD and SDSD are used as emotional stages. Scrolling the window with a specific initial delay can generate an output every second.
3. The precise value of the three parameters is compared with that obtained from the calibration phase in order to increase the Grade of Attention (GA), which is compared with a threshold T_p calculated as:

$$T_p = \frac{3}{2} \cdot WS$$

where 3 represents the number of emotional stages considered.

This comparison produces two levels stored in a Attention Array (AA) from physiological reactions, called AA_p. If the degree of attention is higher than T_p, that window is labeled as *attention*; otherwise, it is labeled as *Distraction*.

4.2.4.3 Merging Physiological and Facial Expressions Indicators

For various reasons, including redundancy and the search for additional useful features, the data relating to AA_{fe} and AA_p were then compared using a windowing algorithm. The objective of the study is not to analyze the attention of individual students but to obtain aggregate results. There are several reasons not to focus on individual students; some are listed below:

- *Privacy*: by aggregating data, the teacher cannot obtain personal feedback from each student's data.
- *Student's interest in the lesson*: it may vary due to factors external to the quality of the teaching itself and personal, including the passion for the topic covered; it may also differ from lesson to lesson for personal reasons, such as drowsiness or mood.
- *Bias*: most datasets on human faces still need to be more balanced due to the difficulty in finding volunteers of different ethnicities. This reduces the prediction performance of AI systems trained using them.

The merged AA obtained from merging both facial expressions and physiological data are defined as $\overline{AA_{fe}}$ and $\overline{AA_p}$.

4.2.5 Lessons

Thirteen participants (four females and nine males, aged 21 to 35) were enrolled in in-person and distance learning lessons to compare attention levels in both modes. Specifically, seven participants were divided into two groups of four and three people, each attending two different lessons, while the remaining six participants participated in the distance learning lessons individually. During the in-person lessons, physiological data were collected via smartwatches and facial expressions via standard cameras operating in the visible wavelength. During the distance learning lessons,

the same variables were collected by adding reaction times, measured using an application designed for this purpose. Both campaigns aimed to assess attention levels and compare results between the two lesson fruition modes.

4.2.5.1 In-Person Lessons

The in-person lessons consisted of a 1-hour lesson regarding how airplanes fly. They involved a teacher and an attendant, who programmed the teacher's wristwatch to send a vibration alarm at predetermined and unknown times. Upon activation of the alarm, the teacher asked the volunteers to perform a simple action, such as touching their shoulder, nose, or head, and whether they were paying attention to the explanation. Reaction times were measured using the video recording of the lesson, starting from the time the teacher requested the action until the volunteers completed it. Before the lessons, one of the two groups was informed of the request to perform the actions, while the second was unaware. In this way, accurate comparisons could be made between the two groups.

4.2.5.2 Distance Learning Lessons

The distance learning lessons consisted of a 50-minute prerecorded lesson regarding the design of an Information Technology (IT) component. They aimed to simulate the online learning environment, which has become increasingly common since the COVID-19 pandemic. Participants were given smartwatches to monitor their physiological reactions during the lesson and were asked to record their facial expressions using a personal device, like a smartphone or a webcam. Since no teacher was present, an application was developed to require the students to confirm whether they were paying attention to the lesson at random periods. Four volunteers were informed about the message windows in advance, while the others were not. This allowed data to be collected on attention levels and reaction times despite the absence of a live teacher.

4.2.5.3 Outcomes

The following results were obtained by comparing the attention levels obtained from both sources in the in-person and distance learning lessons with the volunteers' self-assessments:

- For the in-person lesson, when students were asked if they were paying attention, five out of eight (approximately 62.5%) answered *Yes* every time, while three always answered *No*.
- For the distance learning lessons, each volunteer was asked the same question 5 times using the developed application, resulting in 18 *Yes* and 7 *No* responses after discounting 2 *Yes* answers with excessive reaction times, which were assumed as false positives. This yields an attention level of 72%, measured with a 10-minute window size.

These results differ somewhat from the measurements obtained with the proposed approach (around 52%), but it is challenging to associate statistical relevance due to the small number of volunteers.

4.2.6 Postprocessing

The first challenge was determining an appropriate WS for the windowing algorithm. Since there is limited information in the literature about behavioral analysis on emotional stage parameters, the WS was empirically determined by selecting the value that minimizes the differences between the readouts when the windowing algorithm is applied to both $\overline{AA_{fe}}$ and $\overline{AA_p}$.
Figure 4.4 illustrates the redundancy between the two systems. The Attention Gain (AG) represents the percentage of *Attention* states detected from the raw Attention Behavior (AB), that is, the one to which the 5-minute attention counter is not applied. The graphs show that the best results for the in-person lessons were obtained with a WS of 100 samples. It was decided to use the AGs from the in-person lessons, as the neural network performed better in this case. In the distance learning lessons, there is a strong presence of *neutrality* outcomes, making the results from this source less predictive than the Physiological Data.
Considering 100 samples from both sources, the Root Mean Squared Error (RMSE)

can be calculated using the following formula:

$$RMSE = \sqrt{\frac{(\hat{AG} - AG_p)^2 + (\hat{AG} - AG_{fe})^2}{2}} \tag{4.1}$$

where AG_p and AG_{fe} represent the respective AGs from the physiological data and facial expressions, and $\hat{AG} = \frac{(AG_p + AG_{fe})}{2}$ is the mean between AG_p and AG_{fe}. The RMSE calculated for in-person lessons was 0.02, while that for distance learning lessons was 4.76.

The windowing algorithm results were filtered to analyze the students' collective

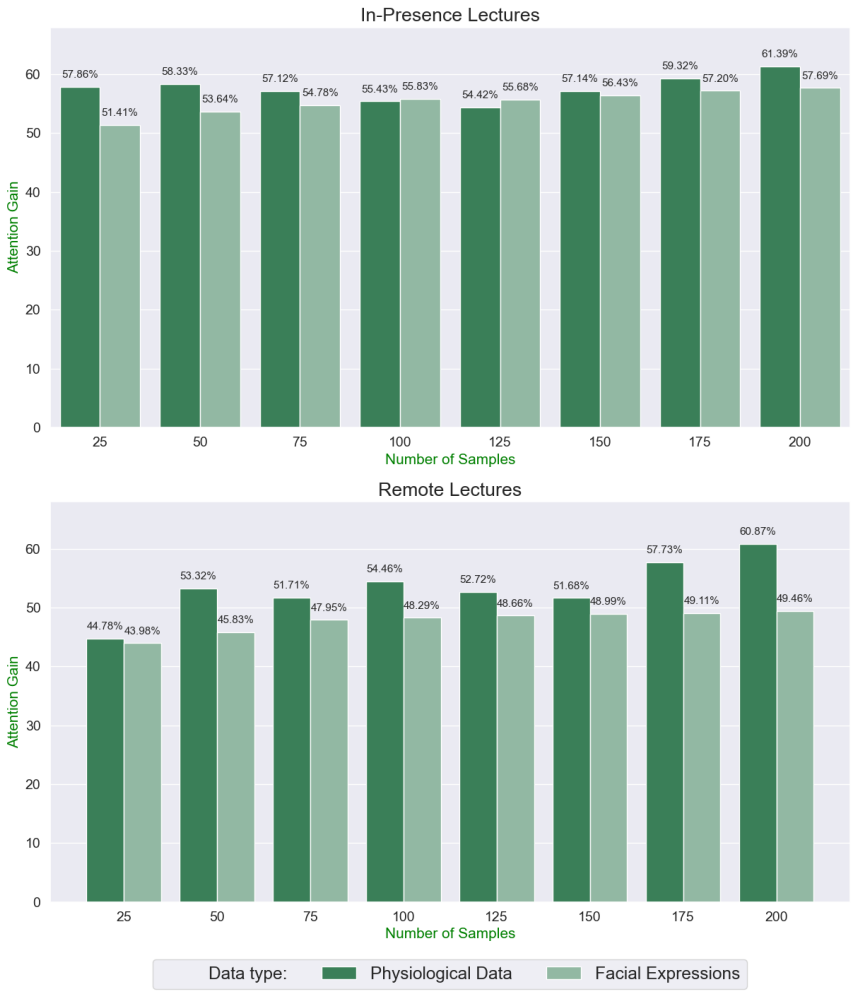

FIGURE 4.4: Comparison of AG calculated with different WS and derived from the $\overline{AA_p}$ and $\overline{AA_{fe}}$ obtained during in-person and distance learning classes. This image was taken from [122].

AB. The results presented in Figure 4.5 were obtained by counting the number of Attention states detected in the previous 5 minutes of the AA_i. This delay does not impact the AB analysis, and since lessons typically last more than an hour, it is

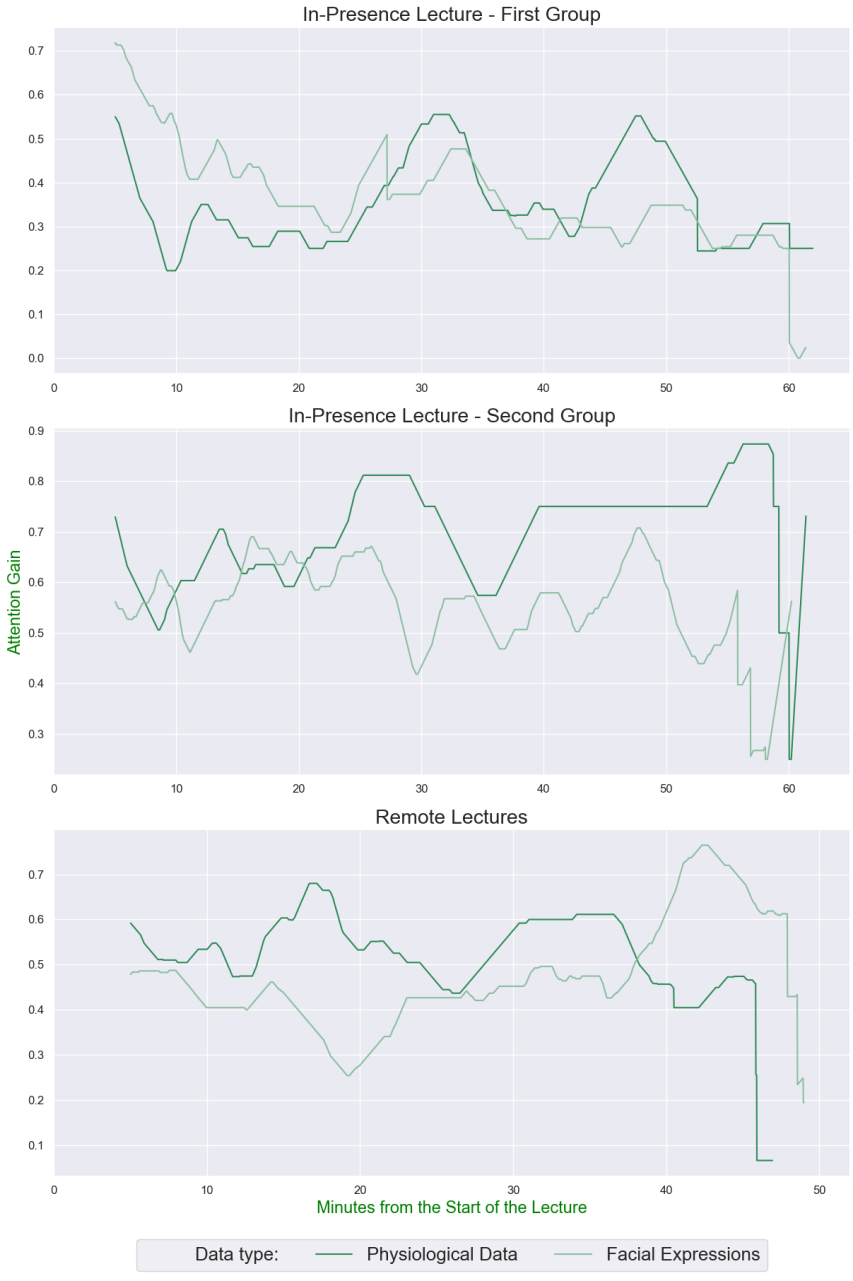

FIGURE 4.5: AB of the in-person and distance learning lessons derived from $\overline{AA_p}$ and $\overline{AA_{fe}}$ with a WS of 100 samples. The 5-minute attention counters are applied every second. This image was taken from [122].

acceptable for a real-world application. Figure 4.6 shows a step-by-step summary of the proposed approach.

4.2.7 Analysis on the Work Done

The study results offer valuable insights into students' attention behaviors in in-person and distance learning lessons. As shown in Section 4.2.5, the two lesson types show noticeable differences in attention levels. Furthermore, the ABs calculated from facial expressions and physiological data sources are consistent. Figure 4.5 illustrates the average attention status of the in-person (first and second group) and distance learning lessons. An attention gap between the two groups is immediately visible, with the average values indicating that the first group paid less attention than the second. For the latter group, it might be reasonable that:

- It paid more attention, knowing that actions might be required during the lesson.
- Was more interested in the lesson topic than the in-person group.

In distance learning lessons, the passive nature of the lesson resulted in a prevalence of neutral facial expressions, making physiological data beneficial for analyzing attention levels. The peak of attention in the in-person lessons, occurring after around 20 minutes for the second group and 30 minutes for the first group, can be explained by a more engaging topic for the audience during those moments. Additionally, specific events in the in-person lessons, such as a student interacting frequently with the teacher or requesting to touch their heads, may have influenced attention

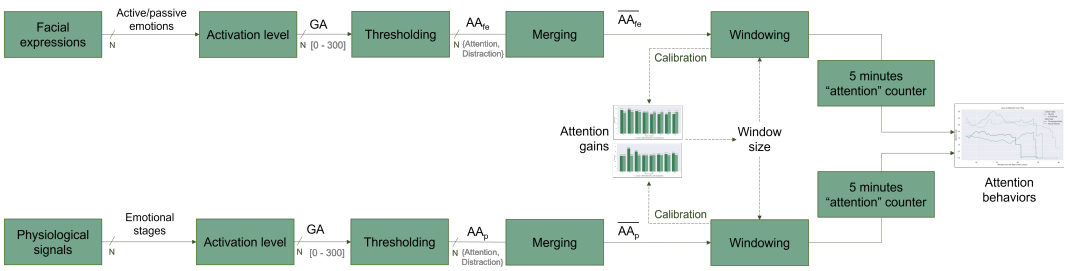

FIGURE 4.6: The workflow of the proposed algorithm. The dotted arrows represent the WS calibration feedback. This image was taken from [122].

levels. Lastly, the results obtained in this study are similar to those found in the two works discussed in Section 4.2, further validating the findings. Reaction times were collected during the experiments but were not used in this work, as no correlation between attention levels and reaction times was found in the literature. The data collection is, however, described due to the potential relevance of the attention checks, which may cause perturbations in student attention levels. Moreover, this information could help replicate the experiments.

Chapter 5

Supervisory Control And Data Acquisition Systems

This chapter discusses a project carried out in collaboration with Sirius [24], a company based in Turin specializing in advanced software systems for power plant management and energy transmission. Data stored by *Supervisory Control And Data Acquisition (SCADA)* systems mounted on *Wind Turbine Generators (WTGs)* was used to identify *anomalous behaviors*, such as those due to thermo-electro-mechanical phenomena [175].

5.1 SCADA Systems

SCADA systems are widely used in telecommunications, water management, energy, oil and gas refining, and transportation [176]. They allow for monitoring and controlling equipment or facilities by transferring data between a central computer and remote devices like Remote Terminal Units (RTUs), Programmable Logic Controllers (PLCs), and operator panels. SCADA systems collect data, relay it to a central hub, and notify relevant personnel. For instance, if a pipeline leaks, the system detects it, sends the data to a central location, analyzes its severity, and presents it clearly for operators to respond.
SCADA systems are of varying complexity, ranging from basic configurations for monitoring the condition of smaller buildings to large-scale installations that enable the management of nuclear power plants or city water systems. In the past, SCADA

systems relied on the Public Switched Network (PSN) to monitor remote equipment. However, many SCADA systems now use corporate Local Area Networks (LANs) or Wide Area Networks (WANs) for monitoring purposes. In addition, wireless technologies are becoming more common for monitoring equipment in remote or hard-to-reach locations. Critical components of a SCADA system include:

1. *Field Devices*: RTUs or PLCs that connect to sensors, switches, and actuators in the field.
2. *Communications Network*: it allows the sending of data between field devices, control units, and the central computer, using mediums like radio, phone, satellite, cable, or their combination.
3. *Central Host Computer*: main server that runs the SCADA system, often called a SCADA Center, Master Station, or Master Terminal Unit (MTU).
4. *Software Systems*: sometimes called Human Machine Interface (HMI) or Man-Machine Interface (MMI) software, they support host computer and operator terminal applications.

5.2 Renewable Energy Sources

Renewable Energy Sources (RESs) are experiencing significant growth globally, driven by efforts to combat climate change and reduce dependence on fossil fuels. According to the International Energy Agency's (IEA) *Renewables 2023* report, RES capacity is projected to increase by $3700GW$ over the next five years [177]. The surge in RES deployment is attributed to initiatives limiting global warming to $1.5°C$, such as the *European Green Deal* launched in 2019 [178, 179]. It aims to transition the EU into a resource-efficient and competitive economy, with measures such as *Fit for 55* and *REPowerEU* aiming to significantly reduce net Greenhouse Gas (GHG) emissions and increase clean energy production by 2030 [180, 181]. Italy, for instance, aims to install around $70GW$ of new renewable capacity under the Fit for 55 plan to cover at least 65% of electricity consumption with clean energy [182]. Geopolitical scenarios related to the Russian-Ukrainian conflict gave additional impetus to this deal to reduce dependence on Russian gas [183].

5.3 Monitoring of Renewable Power Plants

Advanced monitoring systems are crucial to maximize the performance and Return On Investment (ROI) of Renewable Power Plants (RPPs). Without them, detecting faults or suboptimal parameters within the system becomes difficult. Considering that commercial RPPs typically have a lifespan of several decades (for instance, wind farms typically have a lifespan ranging from 20 to 25 years [184]), effective performance analyses and proactive monitoring strategies are essential to minimize OPerational EXpenditure (OPEX). Although the cost of installing a monitoring system can be significant, it is a fundamental investment to maximize the overall yield of the system [185] [186].

Wind energy is one of the most used RES. Monitoring WTGs performance is essential and can be made recurring to various approaches, including *model-driven*, *data-driven*, and *hybrid methods* [187]. Model-based approaches require a deep understanding of the system being modeled. The obtained models, often called *white box*, are mathematically defined and can be understood by anyone with proper preparation and interpretation [188]. Examples of monitoring systems based on these approaches can be found in literature [189–196]. On the other hand, data-driven approaches are preferred for systems that are too complex to model using traditional methods. Models obtained through data-driven approaches, known as *black box*, represent the system without requiring a complete understanding of its behavior. Implementing a data-driven approach typically involves analyzing a significant amount of data to capture the relationship between inputs and outputs [197]. Various proposals have been made following this approach [198–205]. Hybrid approaches combine elements of both model-driven and data-driven methods. These *grey box* models require some system knowledge and utilize a combination of modeling and data analysis techniques [206].

The project discussed in this chapter was carried out in collaboration with Sirius [24] and adopts a *data-driven* approach for identifying anomalous behaviors of WTGs [175]. A novel kind of control chart, defined as *sum-of-events*, is introduced to highlight such anomalous behaviors in the best possible way. One year of 10-minute averaged data collected from WTGs SCADA systems from the same energy player, located in a wind farm in southern Italy, was used.

5.4 Creation of Models

This first phase involves training several Feed-Forward Neural Network (FNN) and Recurrent Neural Network (RNN) models, which aim to represent the healthy behavior of a WTG using preprocessed 10-minute averaged data. The optimal representation of the healthy behavior of a WTG allows highlighting the presence of any anomalous behaviors as they tend to deviate from the training data. The capability of the Artificial Neural Networks (ANNs) to depict the optimal behavior of a WTG is assessed using a different subset of data from the same WTGs.

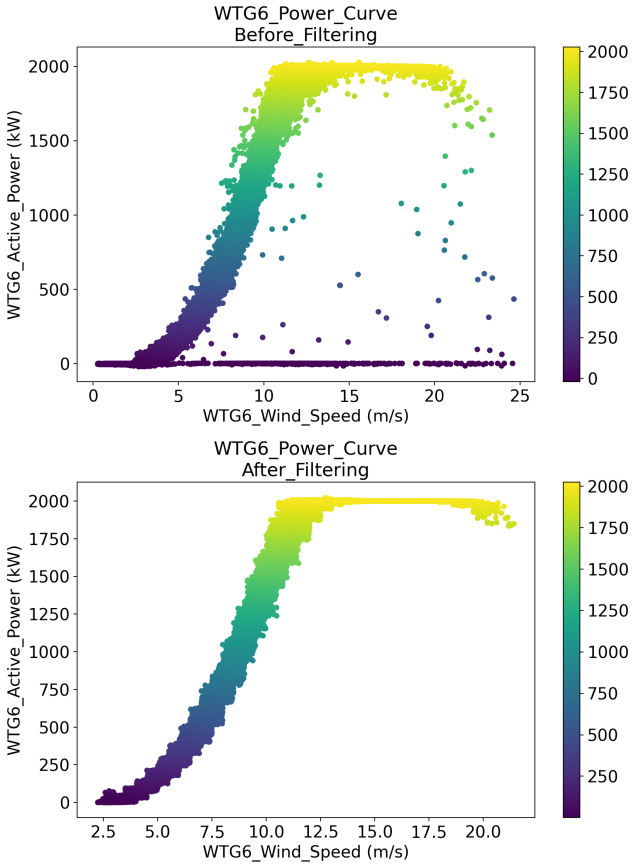

FIGURE 5.1: WTG6 power curve before and after preprocessing procedures.

5.4.1 Preprocessing

Preprocessing is crucial to enable ANNs models to accurately represent the normal operating conditions of the WTGs. As can be seen from Figure 5.1, through preprocessing, the obtained power curve best resembles the ideal power curve of a WTG.

5.4.1.1 Input Data

The initial step involved selecting specific WTGs from the wind farm for investigation. Preference was given to WTG2, WTG6, and WTG7, which had the fewest issues and downtimes. Samples related to known problems and those preceding them were excluded from the analysis. The SCADA data were then processed to ensure consistency and utility throughout all process stages. For each WTG, the following 10-minute average values were extracted:

- Active Power.
- Ambient Temp.
- Blade Pitch Angle.
- Gearbox Bearing Temp.
- Gearbox Oil Temp.
- Generator RPM.
- Generator Temp.
- Rotor RPM.
- SD Wind Speed.
- Wind Speed.
- Wind Direction.

where SD is an acronym for Standard Deviation and RPM is an acronym for Revolutions Per Minute. Additionally, the T-Status was used to assess their operational conditions as it indicates the classified operational status for Reliability, Availability, Maintenance and Safety (RAMS) information about WTGs.

5.4.1.2 Data Cleansing

Invalid values caused by communication errors or inconsistent readings were removed. Consecutive null values in a number greater than 12 were discarded, while those in a smaller amount were chosen to be replaced with the last non-null value.

5.4.1.3 Outlier Filtering

Outliers in the remaining data, indicative of abnormal performances, were managed and potentially removed to represent healthy behavior accurately. Another filtering operation addressed very low power output for high wind speeds. The data binning method, adapted from [207], was used to partition the data based on wind speed intervals of 0.5 m/s. The SD and arithmetic mean of power output values for each interval were calculated to identify and filter out anomalous behavior.

5.4.1.4 Downtime Filtering

All data that was not representative of the average production of a WTG was filtered out to improve the detection of outliers. First of all, periods with power output lower than zero were excluded. Samples corresponding to downtime or failures identified using the T-Status variable were eliminated. Moreover, the data relating to dispatching orders for periods of power reduction were also removed. For specific periods, the WTGs of an entire wind farm may need to run in depowered mode and feed less energy into the grid. These may be due to maintenance needs or constraints imposed by Terna S.p.A., the Transmission System Operator (TSO) grid in Italy.
Subsequently, wind speed was utilized to partition the available data into clusters, or bins, with an interval width of $0.5 m/s$, considering the maximum wind speed in the dataset was $25 m/s$. The SD and arithmetic mean of power output values were computed within each bin. Finally, *control charts*, or *Shewhart charts*, were used to filter additional data. Assuming a normal distribution, the Upper Control Limit (UCL) and Lower Control Limit (LCL) can be defined as follows:

$$UCL : \mu + Z\sigma; \qquad LCL : \mu - Z\sigma \tag{5.1}$$

where σ is the SD and Z represents the distance of the control limits from the mean μ. After several tests, a Z value of 2.5 was chosen for this filtering operation.

5.4.2 Model Definition

The complex nonlinear dependencies between the characteristic parameters of a WTG can be captured using ANNs. The concept of *model set*, a collection of trained

ANN models representing a specific WTG component and its associated physical property, is introduced to facilitate discussion. Table 5.1 overviews the various model sets built to predict specific output variables. Next, the selection of hyperparameters and input sets for these models, their training process, and the resulting results will be discussed.

5.4.2.1 Features Selection for Each Model Set

Selecting the most appropriate set of input variables is crucial for describing the healthy behavior of the WTG system, thus facilitating the detection of anomalous behaviors. In-depth bibliographic research was therefore conducted to ascertain which variables, available among those managed by SCADA systems, are most relevant for implementing a WTG monitoring system, focusing particular attention on the most widespread types of faults [208]. The correlation between output variables and failures tipically relies on engineering expertise.
The optimal input variables identified to predict the output variables of interest were found both using multivariate analysis techniques and carrying out multiple tests. They are presented in Table 5.1.

TABLE 5.1: Outputs, best input set, Z coefficients, and control chart limits for each model set. This table was taken from [175].

Model Set	Output	Best Input Set	Z	UCL, LCL
Wind Turbine	Active Power	Ambient Temp., Wind Speed, STD Wind Speed, Wind Direction	4.4	$+220.01kW$, $-219.68kW$
Gearbox Oil	Gearbox Oil Temp.	Active Power, Ambient Temp., Rotor RPM, Wind Speed, Blade Pitch Angle, Gearbox, Bearing Temp., Generator RPM, Generator Temp.	3.7	$1.44°C$, $-1.37°C$

TABLE 5.2: Model types and description. This table was taken from [175].

Architecture	Input Layer	Hidden Layers
FNN I	A number of neurons equal to the number of input variables	A FC layer with 50 neurons
FNN II	A number of neurons equal to the number of input variables	Two FC layers with 50 and 18 neurons
Single LSTM	A number of neurons equal to the number of timesteps times the number of input variables	A FC layer with 50 neurons
Stacked LSTM	A number of neurons equal to the number of timesteps times the number of input variables	Four FC layers with 64, 50, 25, and 1 neurons

5.4.2.2 Artificial Neural Networks

Four different ANNs were tested for the task at hand:

- Two FNNs consisting exclusively of Fully Connected (FC) layers that use the Rectified Linear Unit (RELU) as the activation function.
- Two RNNs based on the Long Short-Term Memory (LSTM) cell.

Table 5.2 summarizes the selected architectures. Two different types of normalizations were evaluated to improve the learning ability of networks:

- Scaling between 0 and 1.
- Z-Score.

During training, the loss function was monitored using the Mean Squared Error (MSE), while the performance of the models was assessed using the Root Mean Squared Error (RMSE). Several callback functions were implemented for training:

- A Learning Rate (LR) reduction mechanism to be triggered after 4 epochs in which the loss function on the validation set stopped improving. The reduction factor was set at 20% of the current value, to be stopped upon reaching a LR equal to 1% of the initial one.

TABLE 5.3: The list of parameters and hyperparameters used to obtain the best models. This table was taken from [175].

Model Set	Network Type	Hidden Layers and Neurons	Scaling	Min. Val. Loss	Training Epochs	Min. Val. RMSE	Batch Size	Initial LR
Wind Turbine	Stacked LSTM	4 (64,50,25,1)	Z-Score	0.00499 at epoch 16	21	0.05183	64	0.0005
Gearbox Oil	Stacked LSTM	4 (64,50,25,1)	Between 0 and 1	0.00065 at epoch 23	28	0.01508	64	0.0005

TABLE 5.4: Comparison of the metrics obtained by the best models on the WTG2, WTG6, and WTG7 test sets. This table was taken from [175].

Model Set	Output	RMSE WTG2	RMSE WTG6	RMSE WTG7	Avg. RMSE
Wind Turbine	Active Power	48.781kW	48.892kW	52.946kW	50.207kW
Gearbox Oil	Gearbox Oil Temp.	0.543°C	0.306°C	0.283°C	0.377°C

- A termination of the training to be triggered if, after 20 training epochs, there were no improvements in the MSE validation loss.

5.4.2.3 Dataset

The dataset for each WTG was divided into training, validation, and test datasets with a respective subdivision of $80\% - 10\% - 10\%$. The data from WTG2, WTG6, and WTG7 were concatenated in the training and validation datasets. At the same time, the test datasets were kept separate to evaluate the performance of individual WTGs at the end of the training. For FNNs, the prediction of the output at time t depends on the value of the input data at the same time t. Differently, for LSTMs, time series were created using the rolling window principle: the output value at time t depends on the value of each input in the previous two hours.

5.4.3 Training and Testing Analysis

The ANNs under exam were trained on each model set, for each associated input set, and for each data normalization method. Different combinations of hyperparameters were also explored, including batch sizes and LRs. The performance of the trained models was then evaluated on the test datasets of WTG2, WTG6, and WTG7. The best model for each of the four model sets was determined by averaging the RMSE obtained for each WTG. Table 5.3 summarizes the best parameters used to obtain

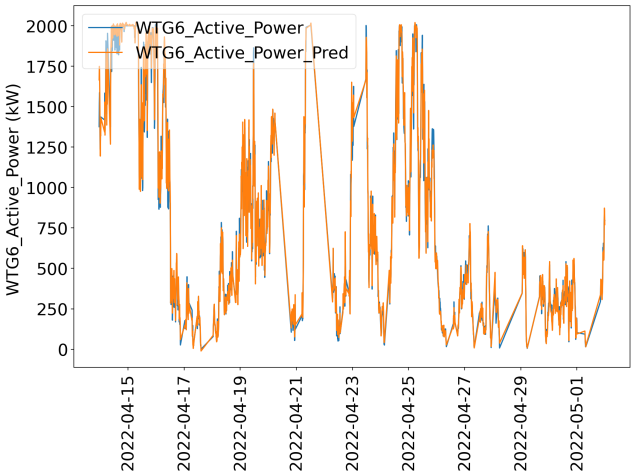

FIGURE 5.2: Comparison of actual and predicted active power values for the WTG6 test set.

the best network for each model set. In contrast, Table 5.4 presents the performance metrics of both best models, including RMSE values calculated on each test set of the three WTGs and their averages. Figure 5.2 compares actual and predicted active power values for the WTG6 test set using the best model trained for the Wind Turbine model set. The model can provide plausible 10-minute predictions, closely following the general trend of active power. Equally good results were obtained for the other sets of models, which were chosen not to show in graphic form for redundancy reasons.

5.4.3.1 Validation

The validation stage is crucial to ensure that the trained models accurately capture the healthy behavior of the analyzed WTGs. The *prediction error*, also known as *deviation* or *delta* (Δ), plays a pivotal role in outlier detection, as large absolute values suggest the presence of outliers. It was computed for each sample using the following formula:

$$\Delta = actual\ value - estimated\ value \tag{5.2}$$

The next step involved merging the delta values from the three test sets and analyzing their distribution to ensure that it resembles a random distribution with a mean value around zero. This analysis helps validate the ability of the models to represent the behavior of the system accurately. Next, parameters were defined for the control

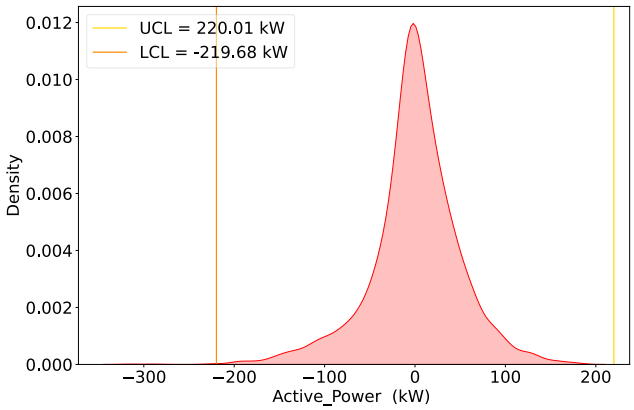

FIGURE 5.3: The distribution of active power delta values of test sets WTG2, WTG6 and WTG7 combined and their respective UCL and LCL.

charts to detect outliers related to new data and other WTGs on the unfiltered WTG datasets. One control chart was used per model, with the LCL and UCL values chosen based on the mean and SD of the delta values. Initially, it was considered a value of $Z = 3$, i.e., the 3σ rule, which states that 99.7% of normally distributed data is within three standard deviations of the mean. Since the test data was already filtered during preprocessing to remove outliers, it was ensured that almost all test values fell within the limits. The Z value was then modified to include up to 99.9% of the values within the UCL and LCL limits. After completing the previous steps, an analysis of the error predictions as a function of the actual output value was conducted to ensure they were distributed across all output data values. This analysis also verified that most fell within the established limits, with only a few values exceeding the UCL and LCL thresholds. Figure 5.3 shows the distribution of prediction errors related to active power for all 3 WTGs considered and the UCL and LCL limits.
The delta/output graph depicted in Figure 5.4 provides valuable insights:

1. It visually represents the delta value of each sample against the corresponding real active power value. This illustration demonstrates that, on average, the deltas are evenly distributed.

2. It confirms that nearly all delta values (99.9%) fall within the specified limits.

Additionally, the error predictions values for each WTG were incorporated into a time series control chart to conduct a time-based evaluation and ensure there were no seasonal anomalies. Upon completing these steps, the model was deemed validated.

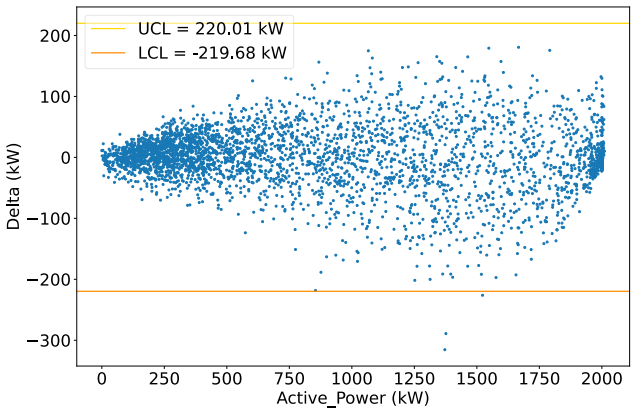

FIGURE 5.4: Graphical representation of the delta value of each sample as a function of the actual active power value. From now on, it will be referred to as delta/output graph.

Figure 5.5 presents the time series control chart for the delta of the active power calculated for WTG6 and the corresponding limits.

5.5 Generalization Ability

This phase aims to verify the generalization ability of the previously trained model sets to different WTGs within the same wind farm, namely WTG1, WTG3, WTG4, and WTG5.

5.5.1 Data Acquisition and Preprocessing

In this phase, more approximate preprocessing was carried out than in Section 5.4.1. The primary assumption is that the previously trained models can describe the healthy behavior of a WTG; therefore, it is wanted to test their ability to recognize anomalous behaviors related to WTGs different from the training ones. Data from multiple WTGs were not concatenated but used individually to calculate individual performance scores. Moreover, only data from inactive periods of the WTGs have been removed to avoid false-negative fault outcomes.

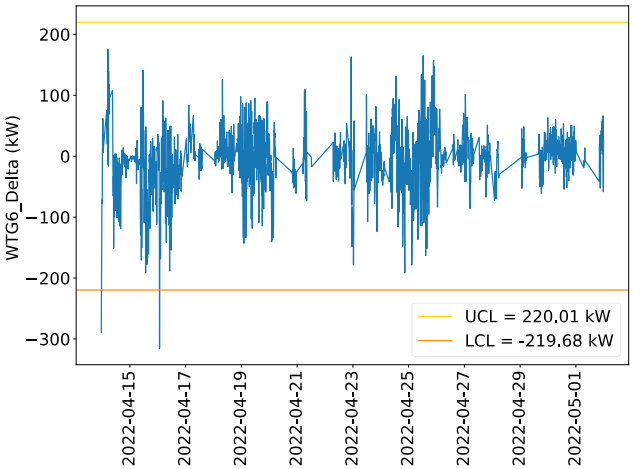

FIGURE 5.5: Time series control chart calculated on the WTG6 test set.

5.5.2 Testing Analysis

As described earlier, all available data for each WTG is utilized as a test set. From Table 5.5, it is evident that the RMSE value calculated on the test sets of WTG1, WTG3, WTG4, and WTG5 is higher than that calculated on WTG2, WTG6, and WTG7. This discrepancy is mainly due to the less rigorous cleaning of these data than the training ones. Figure 5.6 compares the actual active power values and those predicted for the WTG1 test set using the Wind Turbine model set. Only a tiny portion of the dataset, approximately one and a half months, is displayed to maintain visual clarity. The similarity between the two curves suggests that, at least for this variable, generalization to other WTGs may be possible. Similar checks were conducted for the other model sets, yielding positive results.

5.5.3 Sum-Of-Events Control Charts

After predicting the output variables for each model and turbine, the prediction errors were calculated and inserted into the control charts using the LCL and UCL calculated in the previous step. Control charts, while helpful, have only sometimes proven effective in isolating significant outliers and generating alerts. Therefore, an alternative control chart, defined as *sum-of-events*, was developed to group outliers by period. This approach treats outliers as events and splits into positive and negative prediction errors. Positive ones exceed the UCL, while negative ones fall below the LCL. Graphically, this distinction is highlighted with peaks colored red for positive deltas and purple for negative ones. All events from the last 72 hours are summed using a sliding window, and an alarm is generated when there are more than N events in this time interval. The value of N was assigned ten as it was considered an adequate tradeoff. A further analysis of the prediction errors as a function of the actual output variable was conducted through the so-called *delta/output* graph. This analysis helped verify the presence of relationships between the variables.

TABLE 5.5: Comparison of the metrics obtained by the best models on the WTG1, WTG3, WTG4, and WTG5 test sets. This table was taken from [175].

Model Set	RMSE WTG1	RMSE WTG3	RMSE WTG4	RMSE WTG5	Avg. RMSE
Wind Turbine	66.814kW	65.64kW	54.825kW	83.057kW	67.584kW
Gearbox Oil	0.467°C	0.362°C	1.212°C	1.439°C	0.87°C

5.6 Results

The alarm prediction results of each model set are discussed below, along with an analysis of the observed anomalous behaviors.

5.6.1 Wind Turbine

Analysis of the SCADA events revealed malfunctions encountered for WTG5. No significant anomalies were found in the other WTGs. For WTG5, the following alarms were analyzed:

- *Alarm 1*: December 7, 2021, from *02:26:06* to *02:58:24*, for a total duration of 32 minutes and 18 seconds.
- *Alarm 2*: January 27, 2022, from *11:29:35* to *12:34:05*, for a total duration of 1 hour, 4 minutes, and 30 seconds.
- *Alarm 3*: January 27, 2022, from *14:35:20* to *16:26:35*, for a total duration of 1 hour, 51 minutes, and 15 seconds.

All alarms reports the following message: *wind turbine in an emergency state*. While the standard control chart depicted in Figure 5.7 does not indicate anomalous behavior, the sum-of-events type control chart for negative Δs depicted in Figure

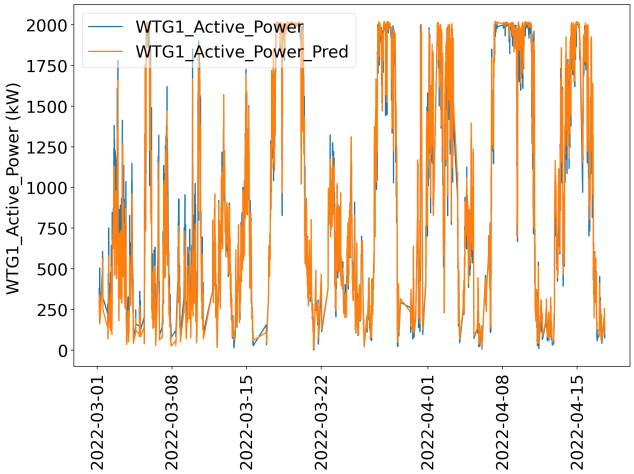

FIGURE 5.6: Comparison of actual and predicted active power values for the WTG1 test set. This image was taken from [175].

5.8 provides a more explicit analysis. The system detected alerts from November 27, 2021, to February 24, 2022, 10 days before Alarm 1. Further analysis of the anomalous values as a function of wind speed was conducted using a specific representation of the power curve. The graph in Figure 5.9 shows predicted and actual active power values, with outliers highlighted separately. During the highlighted period, the WTG appeared depowered, producing up to about $500kW$ less than expected. The delta/output graph reported in Figure 5.10 confirms that the most significant errors occur when the real power produced is between $1.5MW$ and $1.75MW$, which should ideally be greater.

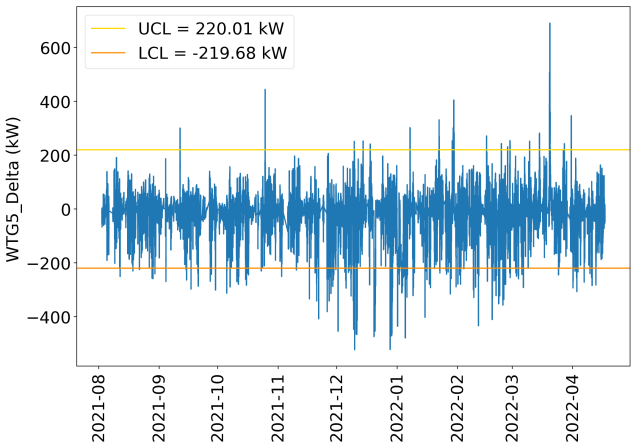

FIGURE 5.7: Error prediction control chart of active power for WTG5.

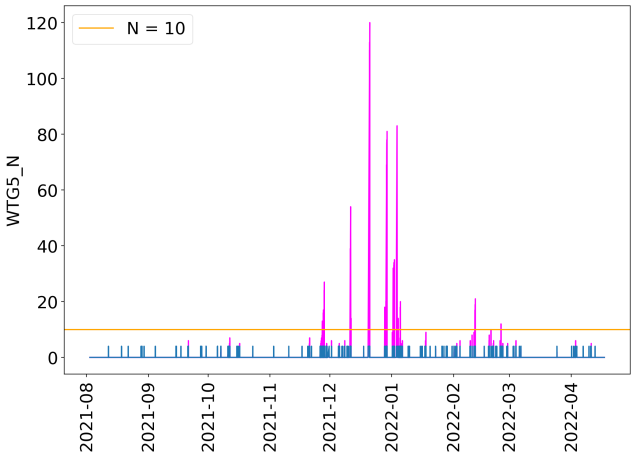

FIGURE 5.8: Sum-of-events type control chart for negative prediction errors calculated for WTG5 active power. This image was taken from [175].

5.6.2 Gearbox Oil

The analysis of SCADA events revealed that WTG5 was affected by a high-temperature alarm, while no significant anomalies were found for the other WTGs. For WTG5, the alarm occurred on August 1, 2021, from 03:17:07 to 10:36:15, for a total duration of 7 hours, 19 minutes, and 8 seconds. The description indicated a high gearbox oil temperature, which exceeded $80°C$ for 1 minute and triggered the fan thermal fuse. The standard control chart depicted in Figure 5.11 shows some anomalous behaviors, but the sum-of-events control chart for positive delta values depicted in Figure 5.12 seems more effective in doing so. In particular, i indicates anomalies starting from August 5, 2021, up to August 6, 2021, and another series of alarms from November 7, 2021. Both graphs show an unusual trend in the gearbox oil temperature between December 10, 2021, and January 4, 2022. This period confirms the presence of a problem with the WTG during those days. The delta/output graph reported in Figure

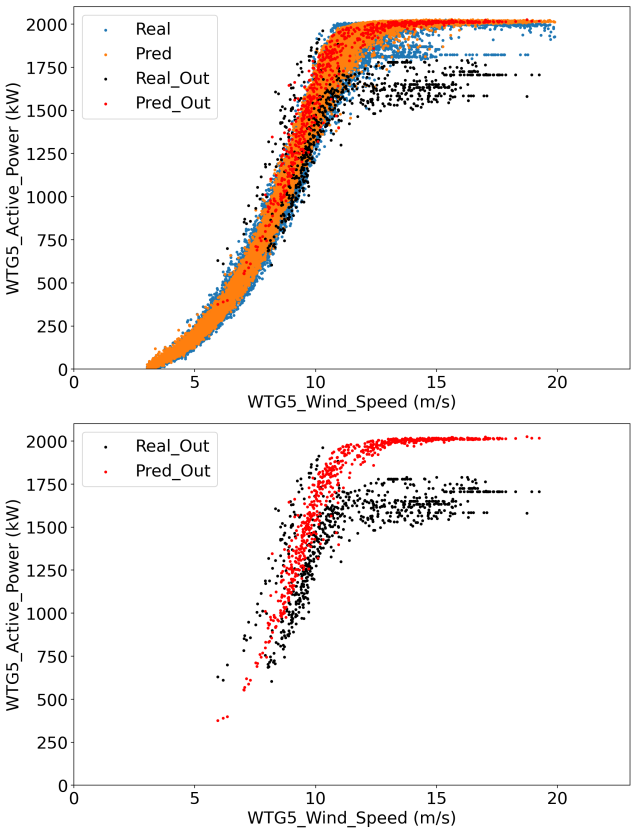

FIGURE 5.9: Power curves for WTG5, with actual (blue) and predicted (orange) values for good predictions, and actual (black) and predicted (red) values for outliers.

5.13 illustrates that these problems occur when the oil temperatures are much higher than the values considered normal. After approximately $74°C$, the delta consistently exceeds the UCL.

5.6.3 Analysis on the Work Done

This Chapter presented a comprehensive system for detecting anomalous behavior in WTGs, divided into two main phases: the Models Definition phase, discussed in Section 5.4.2, and the Generalization Ability phase, discussed in Section 5.5. In the first phase, data from a wind farm in southern Italy were used to train ANNs to represent the healthy status of WTGs. Various preprocessing steps were performed to clean the

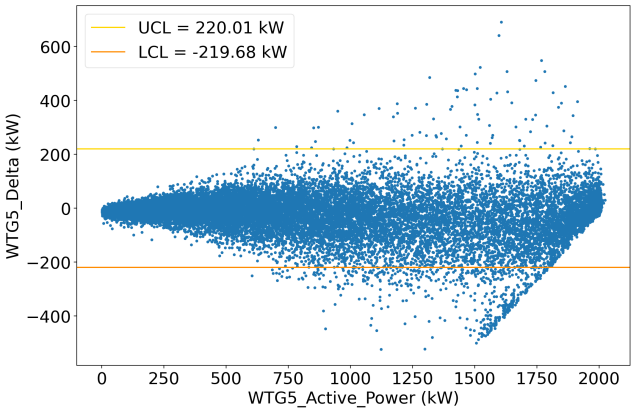

FIGURE 5.10: Delta/output graph of active power for WTG5.

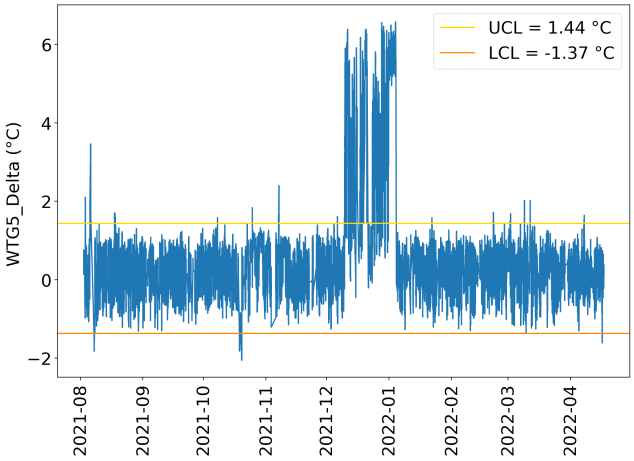

FIGURE 5.11: Error prediction control chart of gearbox oil temperature for WTG5.

data, and different ANN architectures were trained with different hyperparameters and input sets. The best-performing ANN input pairs were selected for each model set. The trained models were first validated using unseen data from the same WTGs used for training and then further tested in the next phase using data from other WTGs in the wind farm. The sum-of-events control chart seemed promising for detecting anomalous behavior on any WTG of the same model within the same wind farm. In the future, the aim is to adapt the system for real-time anomaly detection and improve all aspects of the process, including utilizing more data to account for seasonal variations in behavior.

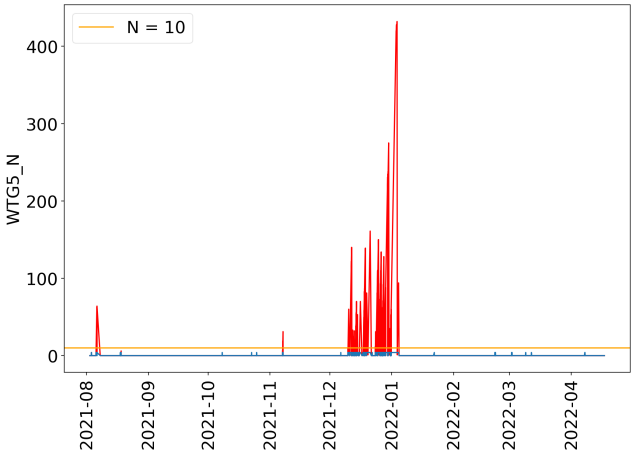

FIGURE 5.12: Sum-of-events type control chart for positive prediction errors values calculated for WTG5 gearbox oil temperature. This image was taken from [175].

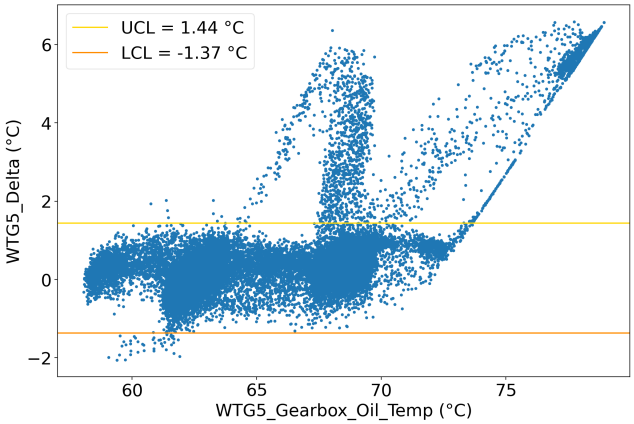

FIGURE 5.13: Delta/output graph of gearbox oil temperature for WTG5.

Chapter 6

Conclusions

This thesis discussed several projects that made use of data coming from different types of sensors. In projects that included the adoption of data-driven philosophies, these data were further analyzed and processed in order to carp and extract their intrinsic value. The choice of classes to be used for training using Machine Learning (ML) techniques involved both the use of techniques derived from multivariate analysis and the search for the physical variables directly involved in the underlying physical process. In the end, multiple contributions were made, which will be listed in the following Sections. Some ideas will also be reported regarding the future direction of the research, which is still continuing.

6.1 Projects Using Fiber Bragg Grating Sensors

This Section will discuss the contributions, limitations, and future work related to projects using fiber Bragg grating sensors.

6.1.1 Contributions

The project using this type of sensor was developed in collaboration with the Department of Mechanical and Aerospace Engineering (DIMEAS) department of the Politecnico di Torino. This was initially supposed to be the main work of my Ph.D. about the research branch. However, the Coronavirus Disease 2019 (COVID-19) crisis forced a change of plan due to distancing regulations and lockdowns imposed by

the national government. The main contribution consists of continuing and bringing to the proof-of-concept stage a work previously started, which concerned the creation of a flying test bench. This required the creation of an open-source software pipeline to overcome the connection limitations of the software originally shipped with the interrogator. Currently, the flying test bench is fully operational and allows the transmission, storage, and visualization of Fiber Bragg Grating (FBG) data even on a Unmanned Aerial Vehicle (UAV) such as Anubis. Multiple software was developed or modified to achieve this scope:

- The Middleware, created using Linux system calls and C/C++, can now transmit sensor data to the Cloud Database. It was described in Section 2.4.3 [35] [36] [37].
- The Three Dimensional (3D) Viewer, developed using Unity and C#, can now receive sensor data from the Cloud Database. It was described in Section 2.4.5.1 [35].
- The FBG Data Analyzer, developed using Qt and C++, is fast and rich in features for visualization. It was described in Section 2.4.5.2 [51].
- The PhotoNext FBG Augmented Reality (AR) Viewer, developed using Unity and C#, leverages the potential of Microsoft Hololens 2 to deliver immersive visualization. It was described in Section 2.4.5.3 [54].

The effectiveness of the pipeline has been demonstrated through numerous laboratory and flight tests described in Sections 2.4.6 and 2.4.7.

6.1.2 Limitations and Future Works

The main limitation of the current work is linked to technology: Fourth Generation (4G) networks, even if they have many advantages in terms of easiness of use and flexibility, can be rather discontinuous and do not allow operation in real-time. To overcome this problem, the research work is proceeding in the following two ways:

- Use of Fifth Generation (5G) networks is being considered. This requires the use of an appropriate Internet Key, which, at the moment, our research team does not possess.

- At the same time, a more rigid pipeline is being evaluated, in which the data is not sent to the Internet from the UAV, but to a ground station from which it is then forwarded to the Internet. This requires creating a new transmission system based on a customized antenna that covers a reasonable transmission distance while ensuring good transmission speed. Although less flexible, this system would allow data to be displayed in real-time at least at the ground station. At the same time, the remote viewers would continue to operate similarly to the current pipeline. Visualizing data in real-time could represent an intrinsic advantage for UAV pilots and other operators.

At the same time, the research work is focusing on other aspects, such as adding an Inertial Measurement Unit (IMU), which could allow the data from the FBG sensors to be correlated with the attitude assumed by the aircraft, the speed, and more. Work is also being done on capturing data relating to the flight commands given by the pilot to the UAV.

6.2 Projects Using Image Sensors

This Section will discuss the contributions, limitations, and future work related to image sensor projects.

6.2.1 Contributions

Projects using image sensors seek to address two issues: combating COVID-19 and improving waste management. When the Ph.D. began, the world was in the midst of the crisis, and most of the Ph.D. work planned before the crisis itself was unfeasible due to distancing regulations and lockdowns. For this reason, the research work focused on other projects that could have been carried out remotely. It was, therefore, decided to use the Information Technology (IT) knowledge of the research group to find solutions to combat the spread of the COVID-19. The effort made for this cause led to the creation of two different datasets with different objectives:

- Facial Mask and Respirator Database (FMR-DB), having as its primary objective the development of automated systems for the recognition of masks and respirators and their typology [68]. It was presented in Section 3.3.1.

- Ways to Wear a Mask or a Respirator Database (WWMR-DB), having as its objective both the recognition of how the mask or respirator is worn and its typology [79]. It was presented in Section 3.3.3.

To have access to the images of the first dataset, it is necessary to sign a fair use research agreement [69], while the second dataset is freely accessible [80].
The second project focused on improving waste management. It was carried out in conjunction with ReLearn, which provided the images used to create an instance segmentation system capable of identifying waste objects within a bin and classifying them according to their constituent material.

6.2.2 Limitations and Future Works

Regarding the first project, the main limitation of the databases created so far is their size. Although the number of images present in the FMR-DB and WWMR-DB is sufficient to train an automated system to recognize the way the mask or respirator is worn and its typology, it could be increased. For this reason, current research focuses on expanding the size of such datasets.
As for the second project, the main limitation is due to the limited number of available images. In the future, efforts will be made to overcome this limitation and further tests will be conducted to improve the performance of the system.

6.3 Project Using Photoplethysmography Sensors

This Section will discuss the contributions, limitations, and future work related to the project using Photoplethysmography (PPG) sensors.

6.3.1 Contributions

So far, only one project has been developed that uses PPG sensors. It was presented in Section 4.2, and it is aimed at aiding teachers in receiving feedback on students' attention levels using non-invasive techniques [122]. Similar systems had been developed previously. However, they used more invasive techniques, such as Electroencephalography (EEG) and Electrocardiogram (ECG), which limited their

effective adoption. Therefore, the main contribution of this research work is to create a similar system that uses less invasive sensors such as a standard image sensor and the PPG sensors with which every smartwatch is equipped. Furthermore, the second objective was to use this data to calibrate the outputs obtained to obtain a system that can operate without PPG sensors, i.e., using images captured exclusively by an image sensor. This leads to a notable reduction in performance but minimizes implementation difficulties (students no longer have to wear any smartwatch) and privacy issues (the algorithm could give a general indication of the attention level of the class and not an individual).

6.3.2 Limitations and Future Works

The limitation of the project above lies mainly in using facial expressions only; from the images, it is possible to trace further characteristics such as gestures, amount of movement, and more. Future research aims to analyze and adopt these characteristics to improve the accuracy of the system created so far.

6.4 Project Using Data From Supervisory Control And Data Acquisition Systems

This Section will discuss the contributions, limitations, and future work related to the project using Supervisory Control And Data Acquisition (SCADA) systems.

6.4.1 Contributions

The project discussed in Section 5.3 and relating to data from a SCADA system aimed at creating an automated system for detecting any anomalous behavior of wind turbines placed in a wind farm in Southern Italy. So far, a complete proof-of-concept has been realized that can detect and highlight abnormal behaviors to any human operators via a control chart called sum-of-events [175].

6.4.2 Limitations and Future Works

At the moment, the project is at an early stage. The obtained results are promising, even though they were carried out using a small subset of wind turbines from a single farm. Therefore, verifying whether the adopted approach can be generalizable to other wind farms and wind turbines is necessary. Furthermore, the sum-of-events control chart must also be examined in depth to understand if it is possible to improve the effectiveness of representing any anomalous behavior. Future work aims to conduct further verifications to improve what has been done so far.

References

[1] Paul J. Crutzen and Eugene F. Stoermer. The 'anthropocene' (2000). In *The Future of Nature*, pages 479–490. Yale University Press, New Haven, 2013.

[2] Theodore Modis. Forecasting the growth of complexity and change. *Technological Forecasting and Social Change*, 69(4):377–404, 2002.

[3] Jonathan Huebner. A possible declining trend for worldwide innovation. *Technological Forecasting and Social Change*, 72(8):980–986, 2005.

[4] Hans P. Moravec. *Robot: Mere Machine to Transcendent Mind*. Oxford University Press USA, 1998.

[5] Ray Kurzweil. The law of accelerating returns. In *Alan Turing: Life and Legacy of a Great Thinker*, pages 381–416. Springer Verlag GmbH, Berlin, Heidelberg, 2004.

[6] Ray Kurzweil. *Age of Spiritual Machines: When Computers Exceed Human Intelligence*. Penguin Books, USA, 1st edition, 1999.

[7] Donncha Kavanagh, Geoff Lightfoot, and Simon Lilley. Are we living in a time of particularly rapid social change? and how might we know? *Technological Forecasting and Social Change*, 169:120856, 2021.

[8] Ronald F. Inglehart, Eduard Ponarin, and Ronald C. Inglehart. Cultural change, slow and fast: The distinctive trajectory of norms governing gender equality and sexual orientation. *Social Forces*, 95(4):1313–1340, 01 2017.

[9] Leda Cosmides and John Tooby. Evolutionary psychology, moral heuristics, and the law. In *Heuristics and the Law*. MIT Press, 2006.

[10] Mark S. Cary. Ad strategy and the stone age brain. *Journal of Advertising Research*, 40(1):103–106, 2000.

[11] Will Steffen, Angelina Sanderson, Peter Tyson, Jill Jäger, Pamela Matson, Berrien Moore, Frank Oldfield, Katherine Richardson, H. Schellnhuber, B. Turner, and Robert Wasson. *Global Change and the Earth System, A Planet Under Pressure*. Springer Verlag GmbH, Berlin, Heidelberg, 01 2005.

[12] Swedish Ministry of Environment. Economy and finance for a just future on a thriving planet. https://financetransformation.earth/wp-content/uploads/2022/05/sthlm50_c1_acceleration_V3.png. [Online]. Accessed on Apr. 30, 2024.

[13] Will Steffen, Wendy Broadgate, Lisa Deutsch, Owen Gaffney, and Cornelia Ludwig. The trajectory of the anthropocene: The great acceleration. *The Anthropocene Review*, 2(1):81–98, 2015.

[14] Kurtis Pykes. The importance of data: 5 top reasons. https://www.datacamp.com/blog/importance-of-data-5-top-reasons. [Online]. Accessed on Apr. 30, 2024.

[15] The Council of Quality and Leadership. 12 reasons why data is important. https://www.c-q-l.org/resources/guides/12-reasons-why-data-is-important/. [Online]. Accessed on Apr. 30, 2024.

[16] Abhishek Singh. What is a sensor? types of sensors, classification & applications. https://www.hackatronic.com/what-is-a-sensor-types-of-sensors-classification-applications/. [Online]. Accessed on Apr. 30, 2024.

[17] Bhagwati Charan Patel, G R Sinha, and Naveen Goel. Introduction to sensors. In *Advances in Modern Sensors*, 2053-2563, pages 1–21. IOP Publishing, 2020.

[18] Jacob Fraden. *Handbook of Modern Sensors: Physics, Designs, and Applications*. Springer, Cham, 2015.

[19] Harry N. Norton. *Handbook of transducers*. Prentice Hall, Englewood Cliffs, 1989.

[20] Ian Robertson Sinclair. *Sensors and Transducers*. Newnes, Oxford, 2000.

[21] Juan Guerrero-Ibáñez, Sherali Zeadally, and Juan Contreras-Castillo. Sensor technologies for intelligent transportation systems. *Sensors*, 18(4), 2018.

[22] Joe F. Hair, William C. Black, Barry J. Babin, and Rolph E. Anderson. *Multivariate Data Analysis*. Prentice Hall, 2009.

[23] Adobe Experience Cloud Team. Multivariate analysis — definition, methods, and examples. https://business.adobe.com/blog/basics/multivariate-analysis-examples#correlation-analysis, 8 2023. [Online]. Accessed on Apr. 30, 2024.

[24] Sirius s.r.l. Sirius | Home. https://www.sirius.to.it/. [Online]. Accessed on Apr. 30, 2024.

[25] ReLearn s.r.l. ReLearn | Home. https://www.re-learn.eu/. [Online]. Accessed on Apr. 30, 2024.

[26] Icarus Student Team. Team icarus polito. https://icarus.polito.it/. [Online]. Accessed on Apr. 30, 2024.

[27] John M. Senior. *Optical Fiber Communications: Principles and Practice*. Pearson, Harlow, 2009.

[28] Gerd Keiser. *Optical Communications Essentials*. McGraw-Hill, New York, 2003.

[29] Hottinger Brüel & Kjær. What is a fiber bragg grating? https://www.hbkworld.com/en/knowledge/resource-center/articles/strain-measurement-basics/optical-strain-sensor-fundamentals/what-is-a-fiber-bragg-grating. [Online]. Accessed on Apr. 30, 2024.

[30] VCELINK. Fiber optic cable types: What you should know. https://www.vcelink.com/blogs/focus/fiber-optic-cable-types. [Online]. Accessed on Apr. 30, 2024.

[31] Slavisa Aleksic. A survey on optical technologies for iot, smart industry, and smart infrastructures. *Journal of Sensor and Actuator Networks*, 8(3), 2019.

[32] Tatsuta Staff. Fiber bragg gratings (fbg). https://www.tatsuta.com/product/sensor_medical/optical/fbg/. [Online]. Accessed on Apr. 30, 2024.

[33] PhotoNext. Photonext. https://www.photonext.polito.it/it. [Online]. Accessed on Apr. 30, 2024.

[34] PhotoNext. Photonext logo. https://media.licdn.com/dms/image/C4E1BAQFQtUJS_vIGpA/company-background_10000/0/1584304093221?e=2147483647&v=beta&t=2WoAk5Mp3AUiYnTrYNhcyTwrOuMx-hpwpPjEFJ21W7A. [Online]. Accessed on Apr. 30, 2024.

[35] Antonio Costantino Marceddu, Gaetano Quattrocchi, Alessandro Aimasso, Edoardo Giusto, Leonardo Baldo, Mohammad Ghazi Vakili, Matteo Davide Lorenzo Dalla Vedova, Bartolomeo Montrucchio, and Paolo Maggiore. Air-to-ground transmission and near real-time visualization of fbg sensor data via cloud database. *IEEE Sensors Journal*, 23(2):1613–1622, 2023.

[36] Antonio Costantino Marceddu and Bartolomeo Montrucchio. Storage and visualization on-the-ground and in near real-time of the data measured by the optical sensors connected to a flying test bench. In *Aerospace Science and Engineering*, volume 33, pages 277–280, 2023.

[37] Antonio Costantino Marceddu and Bartolomeo Montrucchio. Reading, transmission, storage, and display of near real-time data acquired by fiber bragg grating sensors installed on an unmanned aerial vehicle. In *Aerospace Science and Engineering*, volume 42, pages 47–50, 2024.

[38] Antonio Costantino Marceddu, Alessandro Aimasso, Antonio Scaldaferri, Paolo Maggiore, Bartolomeo Montrucchio, and Matteo Davide Lorenzo Dalla Vedova. Creation of a support software for the development of a system for sending and visualizing fbg sensor data for aerospace application. In *2023 IEEE 10th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pages 487–491, 2023.

[39] Mauro Guerrera. Algorithms and methods for fiber bragg gratings sensor networks. Master's thesis, Politecnico di Torino, 2018.

[40] Politecnico di Torino. Photonext emulator. https://github.com/CARDIGANSPoliTo/PhotoNext_Emulator. [Online]. Accessed on Apr. 30, 2024.

[41] Politecnico di Torino. Photonext configurator. https://github.com/CARDIGANSPoliTo/PhotoNext_Configurator. [Online]. Accessed on Apr. 30, 2024.

[42] Politecnico di Torino. Photonext libutils. https://github.com/CARDIGANSPoliTo/PhotoNext_LibUtils. [Online]. Accessed on Apr. 30, 2024.

[43] Politecnico di Torino. Photonext libsmartscan. https://github.com/CARDIGANSPoliTo/PhotoNext_LibSmartScan. [Online]. Accessed on Apr. 30, 2024.

[44] Politecnico di Torino. Photonext middleware. https://github.com/CARDIGANSPoliTo/PhotoNext_Middleware. [Online]. Accessed on Apr. 30, 2024.

[45] Inc. MongoDB. Mongodb. https://www.mongodb.com. [Online]. Accessed on Apr. 30, 2024.

[46] MongoDB. Json and bson. https://www.mongodb.com/json-and-bson. [Online]. Accessed on Apr. 30, 2024.

[47] Unity Technologies. Unity. https://unity.com/. [Online]. Accessed on Apr. 30, 2024.

[48] Maria Giulia Canu. Mixed real-time visualization framework for fgb iot sensors. Master's thesis, Politecnico di Torino, 2019.

[49] Politecnico di Torino. Photonext 3d viewer. https://github.com/CARDIGANSPoliTo/PhotoNext_3D_Viewer. [Online]. Accessed on Apr. 30, 2024.

[50] Sergio Schiavello. Innovative visualization of data generated by photonic sensors. Master's thesis, Politecnico di Torino, 2022.

[51] Antonio Costantino Marceddu, Alessandro Aimasso, Sergio Schiavello, Bartolomeo Montrucchio, Paolo Maggiore, and Matteo Davide Lorenzo Dalla Vedova. Comprehensive visualization of data generated by fiber bragg grating sensors. *IEEE Access*, 11:121945–121955, 2023.

[52] Politecnico di Torino. Photonext fbg data analyzer. https://github.com/CARDIGANSPoliTo/PhotoNext_FBG_Data_Analyzer. [Online]. Accessed on Apr. 30, 2024.

[53] Matteo Davide Lorenzo Dalla Vedova, Pier Carlo Berri, and Alessandro Aimasso. Environmental sensitivity of fiber bragg grating sensors for aerospace prognostics. *Proceedings of the 31st European Safety and Reliability Conference, ESREL 2021*, page 1561 – 1567, 2021.

[54] Antonio Costantino Marceddu, Alessandro Aimasso, Matteo Bertone, Luca Viscanti, Bartolomeo Montrucchio, Paolo Maggiore, and Matteo Davide Lorenzo Dalla Vedova. Augmented reality visualization of fiber bragg grating sensor data for aerospace application. In *2024 IEEE 11th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, 2024.

[55] Takao Kuroda. *Essential Principles of Image Sensors*. CRC Press, 1 2017.

[56] Teledyne DALSA. Ccd vs cmos. https://www.teledynedalsa.com/en/learn/knowledge-center/ccd-vs-cmos/. [Online]. Accessed on Apr. 30, 2024.

[57] Davide Trezzi. Fill factor e full well capacity. https://www.astrotrezzi.it/2017/06/fill-factor-e-full-well-capacity/, 6 2017. [Online]. Accessed on Apr. 30, 2024.

[58] Carlo Dall'Alba. Sensori d'immagine ccd e cmos. Master's thesis, Università degli Studi di Padova, 2012.

[59] MEETOPTICS. Full well capacity. https://www.meetoptics.com/academy/full-well-capacity. [Online]. Accessed on Apr. 30, 2024.

[60] Lucid. Understanding the digital image sensor. https://thinklucid.com/tech-briefs/understanding-digital-image-sensors/. [Online]. Accessed on Apr. 30, 2024.

[61] Teledyne Photometrics. Types of camera sensor. https://www.photometrics.com/learn/camera-basics/types-of-camera-sensor. [Online]. Accessed on Apr. 30, 2024.

[62] Shivali Best. Playstation 5: Huge queues form to buy console despite sony confirming it's not available in-store. https://www.mirror.co.uk/tech/playstation-5-huge-queues-form-23034652. [Online]. Accessed on Apr. 30, 2024.

[63] Dan Sales. Rationing is back! tesco limits flour, eggs, toilet roll and other essentials as shoppers hit oxford street in desperate christmas gift-buying dash before lockdown. https://www.dailymail.co.uk/news/article-8905187/Did-say-LOCKDOWN-Panic-buyers-form-train-trolleys-outside-Sainsburys-Costco.html. [Online]. Accessed on Apr. 30, 2024.

[64] BBC Staff. Covid-19: Crowds flee dhaka ahead of strict bangladesh lockdown. https://www.bbc.com/news/world-asia-57624557. [Online]. Accessed on Apr. 30, 2024.

[65] John Paul Clark. Crowds flock to edinburgh christmas market despite nicola sturgeon's 'stay at home' warning. https://www.edinburghlive.co.uk/news/crowds-flock-edinburgh-christmas-market-22502541. [Online]. Accessed on Apr. 30, 2024.

[66] Thang Pham, Bao Tran, Duy Pham, and Long Nguyen. Classifying between masked faces and normal faces with cnn and ssh. https://github.com/aome510/Mask-Classifier/blob/master/paper/paper.pdf, 8 2019. [Online]. Accessed on Apr. 30, 2024.

[67] Shaohui Lin, Ling Cai, Xianming Lin, and Rongrong Ji. Masked face detection via a modified lenet. *Neurocomputing*, 218:197–202, 2016.

[68] Antonio Costantino Marceddu and Bartolomeo Montrucchio. Recognizing the type of mask or respirator worn through a cnn trained with a novel database. In *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1490–1495, 2021.

[69] Antonio Costantino Marceddu and Bartolomeo Montrucchio. Facial masks and respirators database (fmr-db). *IEEE DataPort*, 2020.

[70] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[71] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I, 2001.

[72] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[73] François Chollet et al. Keras. https://keras.io, 2015.

[74] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16, page 265–283, USA, 2016. USENIX Association.

[75] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[76] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[77] NVIDIA. Nvidia cuda toolkit. https://developer.nvidia.com/cuda-toolkit. [Online]. Accessed on Apr. 30, 2024.

[78] NVIDIA. Nvidia cuda deep neural network library (cudnn). https://developer.nvidia.com/cudnn. [Online]. Accessed on Apr. 30, 2024.

[79] Antonio Costantino Marceddu, Renato Ferrero, and Bartolomeo Montrucchio. Mask and respirator detection: analysis and potential solutions for a frequently ill-conditioned problem. In *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1056–1061, 2022.

[80] Antonio Costantino Marceddu and Bartolomeo Montrucchio. Ways to wear a mask or a respirator (wwmr-db). *IEEE DataPort*, 2021.

[81] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[82] Adnane Cabani, Karim Hammoudi, Halim Benhabiles, and Mahmoud Melkemi. Maskedface-net – a dataset of correctly/incorrectly masked face images in the context of covid-19. *Smart Health*, 19:100144, 2021.

[83] Borut Batagelj, Peter Peer, Vitomir Štruc, and Simon Dobrišek. How to correctly detect face-masks for covid-19 from visual information? *Applied Sciences*, 11(5), 2021.

[84] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010.

[85] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society.

[86] Tzutalin et al. Labelimg. https://github.com/HumanSignal/labelImg?tab=readme-ov-file, 2020. [Online]. Accessed on Apr. 30, 2024.

[87] Humans In The Loop. Medical mask dataset. https://humansintheloop.org/medical-mask-dataset/, 2020. [Online]. Accessed on Apr. 30, 2024.

[88] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 43(12):4217–4228, dec 2021.

[89] Silpa Kaza, Lisa C. Yao, Perinaz Bhada-Tata, and Frank Van Woerden. *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*. The World Bank Group, Washington, DC, 9 2018.

[90] Eugènia Mariné Barjoan, Nadège Doulet, Amel Chaarana, Julie Festraëts, Agnès Viot, Damien Ambrosetti, Jean-Luc Lasalle, Nicolas Mounier, Laurent Bailly, and Christian Pradier. Cancer incidence in the vicinity of a waste incineration plant in the nice area between 2005 and 2014. *Environmental Research*, 188:109681, 2020.

[91] Yushuo Chang. Waste detection based on mask r-cnn. Master's thesis, Politecnico di Torino, 2022.

[92] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[93] Abhishek Dutta, Ankush Gupta, and Andrew Zisserman. VGG image annotator (VIA). http://www.robots.ox.ac.uk/ vgg/software/via/, 2016. [Online]. Accessed on Apr. 30, 2024.

[94] Abhishek Dutta and Andrew Zisserman. The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, New York, NY, USA, 2019. ACM.

[95] Sriram Govardhanam. wastedata-multiple-classes. https://github.com/SriRamGovardhanam/wastedata-Mask_RCNN-multiple-classes, 2020. [Online]. Accessed on Apr. 30, 2024.

[96] Luigi Pugliese, Massimo Violante, and Sara Groppo. A novel algorithm for detecting the drowsiness onset in real-time. *IEEE Access*, 10:42601–42606, 2022.

[97] Muhammad Shafique and Sandip Pal. Investigation of photoplethysmographic signals and blood oxygen saturation values on healthy volunteers during cuff-induced hypoperfusion using a multimode PPG/SpO 2 sensor. *Medical & biological engineering & computing*, 50(6):575–83, 05 2012.

[98] John Allen. Photoplethysmography and its application in clinical physiological measurement. physiological measurement. *Physiological Measurement*, 28(3):1–39, Feb 2007.

[99] Elisa Mejía-Mejía, John Allen, Karthik Budidha, Chadi El Hajj, Panicos Kyriacou, and Peter Charlton. *Photoplethysmography signal processing and synthesis*, pages 69–146. Academic Press, 01 2022.

[100] Karthik Budidha. In vivo investigation of ear canal pulse oximetry during hypothermia. *Journal of Clinical Monitoring and Computing*, 32:97–107, 02 2018.

[101] Subhasri Chatterjee and Panayiotis A. Kyriacou. Monte carlo analysis of optical interactions in reflectance and transmittance finger photoplethysmography. *Sensors*, 19(4), 2019.

[102] Subhasri Chatterjee, Karthik Budidha, and Panayiotis A Kyriacou. Investigating the origin of photoplethysmography using a multiwavelength monte carlo model. *Physiological Measurement*, 41(8):084001, sep 2020.

[103] Andreia Moço, Sander Stuijk, and Gerard Haan. New insights into the origin of remote PPG signals in visible light and infrared. *Scientific Reports*, 8, 05 2018.

[104] Alexei A. Kamshilin and Nikita B. Margaryants. Origin of photoplethysmographic waveform at green light. *Physics Procedia*, 86:72–80, 2017.

[105] Peter Charlton. Photoplethysmogram signal components. https://commons.wikimedia.org/wiki/File:Photoplethysmogramsignalcomponents.svg. [Online]. Accessed on Apr. 30, 2024.

[106] Peter Charlton. Photoplethysmogram pulse wave. https://commons.wikimedia.org/wiki/File:Photoplethysmogram_(PPG)_pulse_wave.svg. [Online]. Accessed on Apr. 30, 2024.

[107] Sandrine C. Millasseau, Franck G. Guigui, Ronan P. Kelly, Krishna Prasad, John R. Cockcroft, James M. Ritter, and Philip J. Chowienczyk. Noninvasive assessment of the digital volume pulse. *Hypertension*, 36(6):952–956, 2000.

[108] Mohamed Elgendi. On the analysis of fingertip photoplethysmogram signals. *Current Cardiology Reviews*, 8(1):14–25, 2012.

[109] Thomas R. Dawber, H. Emerson Thomas Jr., and Patricia M. McNamara. Characteristics of the dicrotic notch of the arterial pulse wave in coronary heart disease. angiology. *Angiology*, 24(4):244–255, 04 1973.

[110] Peter Charlton, Patrick Celka, Bushra Farukh, Phil Chowienczyk, and Jordi Alastruey. Assessing mental stress from the photoplethysmogram: A numerical study. *Physiological Measurement*, 39, 04 2018.

[111] C. El-Hajj and P.A. Kyriacou. A review of machine learning techniques in photoplethysmography for the non-invasive cuff-less measurement of blood pressure. *Biomedical Signal Processing and Control*, 58:101870, 2020.

[112] Mohamed Elgendi. *PPG Signal Analysis An Introduction Using MATLAB*. CRC Press, Boca Raton, 12 2020.

[113] Fred Shaffer and J. P. Ginsberg. An overview of heart rate variability metrics and norms. *Frontiers in Public Health*, 5, 2017.

[114] Rollin McCraty and Fredric Shaffer. Heart rate variability: New perspectives on physiological mechanisms, assessment of self-regulatory capacity, and health risk. *Global Advances in Integrative Medicine and Health*, 4:46–61, 02 2015.

[115] Ary Goldberger. Is the normal heartbeat chaotic or homeostatic? *News in Physiological Sciences*, 6:87–91, 04 1991.

[116] Richard N. Gevirtz, Paul M. Lehrer, and Mark S. Schwartz. *Cardiorespiratory Biofeedback*, chapter 13, page 196–213. The Guilford Press, 2016.

[117] Frank Beckers, Bart Verheyden, and André Aubert. Aging and nonlinear heart rate control in a healthy population. *American journal of physiology. Heart and circulatory physiology*, 290:H2560–70, 07 2006.

[118] Phyllis Stein, Peter Domitrovich, Nelson Hui, Pentti Rautaharju, and John Gottdiener. Sometimes higher heart rate variability is not better heart rate variability: Results of graphical and nonlinear analyses. *Journal of cardiovascular electrophysiology*, 16:954–9, 09 2005.

[119] Richard David Lane, Eric M. Reiman, Geoffrey L. Ahern, and Julian F. Thayer. Activity in medial prefrontal cortex correlates with vagal component of heart rate variability during emotion. *Brain and Cognition*, 47:97–100, 01 2001.

[120] Mika P. Tarvainen, Jukka Lipponen, Juha-Pekka Niskanen, and Perttu O. Ranta-Aho. Hrv version 3.1 – user's guide. https://www.kubios.com/downloads/Kubios_HRV_Users_Guide_3_1_0.pdf, 2018. [Online]. Accessed on Apr. 30, 2024.

[121] Phyllis Stein and Anand Reddy. Non-linear heart rate variability and risk stratification in cardiovascular disease. *Indian pacing and electrophysiology journal*, 5:210–20, 02 2005.

[122] Antonio Costantino Marceddu, Luigi Pugliese, Jacopo Sini, Gustavo Ramirez Espinosa, Mohammadreza Amel Solouki, Pietro Chiavassa, Edoardo Giusto, Bartolomeo Montrucchio, Massimo Violante, and Francesco De Pace. A novel redundant validation IoT system for affective learning based on facial expressions and biological signals. *Sensors*, 22(7), 2022.

[123] Arindam Ray and Amlan Chakrabarti. Design and implementation of technology enabled affective learning using fusion of bio-physical and facial expression. *Journal of Educational Technology & Society*, 19(4):112–125, 2016.

[124] Hamed Monkaresi, Nigel Bosch, Rafael A. Calvo, and Sidney K. D'Mello. Automated detection of engagement using video-based estimation of facial expressions and heart rate. *IEEE Transactions on Affective Computing*, 8(1):15–28, 2017.

[125] Renate Nummela Caine and Geoffrey Caine. *Making connections: teaching and the human brain*. Association for Supervision and Curriculum Development, 1991.

[126] Evan F. Risko, Nicola Anderson, Amara Sarwal, Megan Engelhardt, and Alan Kingstone. Everyday attention: variation in mind wandering and memory in a lecture. *Applied Cognitive Psychology*, 26(2):234–242, 2012. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/acp.1814.

[127] Evan F. Risko, Dawn Buchanan, Srdan Medimorec, and Alan Kingstone. Everyday attention: mind wandering and computer use during lectures. *Computers & Education*, 68:275–283, 2013.

[128] Byron A. Campbell, Harlene Hayne, and Rick Richardson. *Attention and information processing in infants and adults: perspectives from human and animal research*. Psychology Press, 2014.

[129] Karen Wilson and James H. Korn. Attention during lectures: beyond ten minutes. *Teaching of Psychology*, 34(2):85–89, 2007.

[130] Ali Darvishi, Hassan Khosravi, Shazia Sadiq, and Barbara Weber. Neurophysiological measurements in higher education: a systematic literature review. *International Journal of Artificial Intelligence in Education*, 2021.

[131] Jiahui Xu and Baichang Zhong. Review on portable EEG technology in educational research. *Computers in Human Behavior*, 81:340–349, 2018.

[132] Genaro Rebolledo-Mendez, Ian Dunwell, Erika A. Martínez-Mirón, María Dolores Vargas-Cerdán, Sara de Freitas, Fotis Liarokapis, and Alma R. García-Gaona. Assessing neurosky's usability to detect attention levels in an assessment exercise. In Julie A. Jacko, editor, *Human-Computer Interaction. New Trends*, Lecture Notes in Computer Science, pages 149–158, Berlin, Heidelberg, 2009. Springer.

[133] Bartolomeo Montrucchio, Edoardo Giusto, Mohammad Ghazi Vakili, Stefano Quer, Renato Ferrero, and Claudio Fornaro. A densely-deployed, high sampling rate, open-source air pollution monitoring WSN. *IEEE Transactions on Vehicular Technology*, 69(12):15786–15799, 2020.

[134] Luigi Atzori, Roberto Girau, Salvatore Martis, Virginia Pilloni, and Marco Uras. A SIoT-aware approach to the resource management issue in mobile crowdsensing. In *2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN)*, pages 232–237, 2017. issn: 2472-8144.

[135] Luigi Atzori, Roberto Girau, Virginia Pilloni, and Marco Uras. Assignment of sensing tasks to IoT devices: exploitation of a social network of objects. *IEEE Internet of Things Journal*, 6(2):2679–2692, 2019. Conference Name: IEEE Internet of Things Journal.

[136] Luigi Atzori, Claudia Campolo, Bin Da, Roberto Girau, Antonio Iera, Giacomo Morabito, and Salvatore Quattropani. Enhancing identifier/locator splitting through social internet of things. *IEEE Internet of Things Journal*, 6(2):2974–2985, 2019.

[137] Luigi Atzori, Claudia Campolo, Bin Da, Roberto Girau, Antonio Iera, Giacomo Morabito, and Salvatore Quattropani. Smart devices in the social loops: criteria and algorithms for the creation of the social links. *Future Generation Computer Systems*, 97:327–339, 2019.

[138] Xieling Chen, Di Zou, Haoran Xie, and Fu Lee Wang. Past, present, and future of smart learning: a topic-based bibliometric analysis. *International Journal of Educational Technology in Higher Education*, 18(1):2, 2021.

[139] Ping Tan, Han Wu, Peng Li, and He Xu. Teaching management system with applications of RFID and IoT technology. *Education Sciences*, 8(1), 2018.

[140] Mohammad Ali and Al Maruf Hassan. Developing applications for voice enabled IoT devices to improve classroom activities. In *2018 21st International Conference of Computer and Information Technology (ICCIT)*, pages 1–4, 2018.

[141] Ghazal Yadav, Prabha Sundaravadivel, and Lokeshwar Kesavan. Affect-learn: An IoT-based affective learning framework for special education. In *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, pages 1–5, 2020.

[142] Muhammad Awais, Mohsin Raza, Nishant Singh, Kiran Bashir, Umar Manzoor, Saif Ul Islam, and Joel J. P. C. Rodrigues. Lstm-based emotion detection using physiological signals: IoT framework for healthcare and distance learning in covid-19. *IEEE Internet of Things Journal*, 8(23):16863–16871, 2021.

[143] Paul Ekman and Wallace Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17:124–9, 02 1971.

[144] Paul Ekman. *Basic Emotions*, chapter 3, pages 45–60. John Wiley & Sons, Ltd, 1999.

[145] Daniel T. Cordaro, Rui Sun, Dacher Keltner, Shanmukh Vasant Kamble, Niranjan Huddar, and Galen D McNeil. Universals and cultural variations in 22 emotional expressions across five cultures. *Emotion*, 18:75–93, 2018.

[146] Paul Ekman and Wallace V. Friesen. *Facial action coding system: A technique for the measurement of facial movement*. Consulting Psychologists Press, Palo Alto, CA, 1978.

[147] Maja Pantic and Léon Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proceedings of the IEEE*, 91:1370–1390, 2003.

[148] Y.-I. Tian, T. Kanade, and J.F. Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):97–115, 2001.

[149] Evangelos Sariyanidi, Hatice Gunes, and Andrea Cavallaro. Automatic analysis of facial affect: a survey of registration, representation, and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(6):1113–1133, 2015.

[150] I.A. Essa and A.P. Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):757–763, 1997.

[151] Kenji Mase. Recognition of Facial Expression from Optical Flow. *IEICE TRANSACTIONS on Information and Systems*, E74-D(10):3474–3483, 1991.

[152] Zhentao Liu, Min Wu, Weihua Cao, Luefeng Chen, Jianping Xu, Ri Zhang, Mengtian Zhou, and Junwei Mao. A facial expression emotion recognition based human-robot interaction system. *IEEE/CAA Journal of Automatica Sinica*, 4(4):668–676, 2017.

[153] Pedro M. Ferreira, Filipe Marques, Jaime S. Cardoso, and Ana Rebelo. Physiological inspired deep neural networks for emotion recognition. *IEEE Access*, 6:53930–53943, 2018.

[154] Si Miao, Haoyu Xu, Zhenqi Han, and Yongxin Zhu. Recognizing facial expressions using a shallow convolutional neural network. *IEEE Access*, 7:78000–78011, 2019. Conference Name: IEEE Access.

[155] Ping Liu, Shizhong Han, Zibo Meng, and Yan Tong. Facial expression recognition via a boosted deep belief network. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014.

[156] Zibo Meng, Ping Liu, Jie Cai, Shizhong Han, and Yan Tong. Identity-aware convolutional neural network for facial expression recognition. In *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pages 558–565, 2017.

[157] Salah Rifai, Yoshua Bengio, Aaron Courville, Pascal Vincent, and Mehdi Mirza. Disentangling factors of variation for facial expression recognition. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, Lecture Notes in Computer Science, pages 808–822, Berlin, Heidelberg, 2012. Springer.

[158] Jacopo Sini, Antonio Costantino Marceddu, and Massimo Violante. Automatic emotion recognition for the calibration of autonomous driving functions. *518*, 9(3), 2020.

[159] Antonio Costantino Marceddu. Automatic recognition and classification of passengers' emotions in autonomous driving vehicles. Master's thesis, Politecnico di Torino, Torino, October 2019.

[160] Jacopo Sini, Antonio Costantino Marceddu, Massimo Violante, and Riccardo Dessì. Passengers' emotions recognition to improve social acceptance of autonomous driving vehicles. In Anna Esposito, Marcos Faundez-Zanuy, Francesco Carlo Morabito, and Eros Pasero, editors, *Progresses in Artificial Intelligence and Neural Systems*, pages 25–32, Singapore, 2021. Springer Singapore.

[161] Antonio Costantino Marceddu, Jacopo Sini, Massimo Violante, and Bartolomeo Montrucchio. A novel approach to improve the social acceptance of autonomous driving vehicles by recognizing the emotions of passengers. In Wolfgang Osten, Dmitry P. Nikolaev, and Jianhong Zhou, editors, *Thirteenth International Conference on Machine Vision*, volume 11605, pages 503 – 510. International Society for Optics and Photonics, SPIE, 2021.

[162] Antonio Costantino Marceddu, Jacopo Sini, Bartolomeo Montrucchio, and Massimo Violante. Use of facial expressions to improve the social acceptance of level 4 and 5 automated driving system equipped vehicles. In *2022 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–3, 2022.

[163] Takeo Kanade, Jeffrey F. Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 46–53, 2000.

[164] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 94–101, 2010.

[165] Natalie Ebner, Michaela Riediger, and Ulman Lindenberger. FACES—a database of facial expressions in young, middle-aged, and older women and men: Development and validation. *Behavior research methods*, 42:351–62, 02 2010.

[166] Ian Goodfellow, Dumitru Erhan, Pierre Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, and Y. Bengio. Challenges in representation learning: A report on three machine learning contests. *Neural Networks*, 64, 07 2013.

[167] Emad Barsoum, Cha Zhang, Cristian Canton Ferrer, and Zhengyou Zhang. Training deep networks for facial expression recognition with crowd-sourced label distribution. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI '16, page 279–283, New York, NY, USA, 2016. Association for Computing Machinery.

[168] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expressions with gabor wavelets. In *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 200–205, 1998.

[169] Niki Aifanti, Christos Papachristou, and Anastasios Delopoulos. The MUG facial expression database. In *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10*, pages 1–4, 2010.

[170] Oliver Langner, Ron Dotsch, Gijsbert Bijlstra, Daniel H. J. Wigboldus, Skyler T. Hawk, and Ad van Knippenberg. Presentation and validation of the radboud faces database. *Cognition and Emotion*, 24(8):1377–1388, 2010.

[171] Abhinav Dhall, Roland Goecke, Simon Lucey, and Tom Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 2106–2112, 2011.

[172] G. Bradski. The OpenCV library. *Dr. Dobb's Journal of Software Tools*, 2000.

[173] Eclipse Deeplearning4j Development Team. Deeplearning4j: Open-source distributed deep learning for the jvm, apache software foundation license 2.0. https://deeplearning4j.konduit.ai/. [Online]. Accessed on Apr. 30, 2024.

[174] James Russell. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39:1161–1178, 12 1980.

[175] Antonio Costantino Marceddu, Pier Paolo Politi, Matteo Di Salvo, Fabio Bima, and Bartolomeo Montrucchio. Sum-of-events type control charts for the visual representation of abnormal behavior predictions of wind turbine generators. In Anna Esposito, Gennaro Cordasco, Marcos Faundez-Zanuy, Francesco Carlo Morabito, and Eros Pasero, editors, *Smart Innovation, Systems and Technologies*. Springer, 2024.

[176] NCS TIB. 04-1; technical information bulletin 04–1, supervisory control and data acquisition (scada) systems. *National Communications System: Arlington, VA, USA*, 2004.

[177] International Energy Agency. Renewables 2023. https://www.iea.org/reports/renewables-2023. [Online]. Accessed on Apr. 30, 2024.

[178] Priyadarshi R. Shukla, Jim Skea, Raphael Slade, Alaa Al Khourdajie, Renée van Diemen, David McCollum, Minal Pathak, Shreya Some, Purvi Vyas, Roger Fradera, Malek Belkacemi, Apoorva Hasija, Géninha Lisboa, Sigourney Luz, and Juliette Malley, editors. *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, Cambridge, UK and New York, NY, USA, 2022.

[179] European Commission. Document 52019DC0640. https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1588580774040&uri=CELEX%3A52019DC0640. [Online]. Accessed on Apr. 30, 2024.

[180] European Commission. Document 52021DC0550. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52021DC0550. [Online]. Accessed on Apr. 30, 2024.

[181] European Commission. Document 52022DC0230. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=COM%3A2022%3A230%3AFIN&qid=1653033742483. [Online]. Accessed on Apr. 30, 2024.

[182] Terna Staff. Econnextion: la mappa delle connessioni rinnovabili. https://www.terna.it/it/sistema-elettrico/rete/econnextion. [Online]. Accessed on Apr. 30, 2024.

[183] International Energy Agency. Renewable power's growth is being turbocharged as countries seek to strengthen energy security. https://www.iea.org/news/renewable-power-s-growth-is-being-turbocharged-as-countries-seek-to-strengthen-energy-security. [Online]. Accessed on Apr. 30, 2024.

[184] TWI Staff. How Long do Wind Turbines Last? Can their Lifetime be Extended? https://www.twi-global.com/technical-knowledge/faqs/how-long-do-wind-turbines-last. [Online]. Accessed on Apr. 30, 2024.

[185] Renato Lazzarin. The importance of monitoring renewable energy plants: Three case histories. *Thermal Science and Engineering Progress*, 4:197–204, 2017.

[186] Valerio Mariani, Giovanna Adinolfi, Amedeo Buonanno, Roberto Ciavarella, Antonio Ricca, Vincenzo Sorrentino, G. Graditi, and Maria Valenti. A survey on anomalies and faults that may impact the reliability of renewable-based power systems. *Preprints 2024*, 05 2024.

[187] Silvia Estrada-Flores, Inge Merts, Bart De Ketelaere, and Jeroen Lammertyn. Development and validation of "grey-box" models for refrigeration applications: A review of key concepts. *International Journal of Refrigeration*, 29(6):931–946, 2006.

[188] Xiang Gong and Wei Qiao. Simulation investigation of wind turbine imbalance faults. In *2010 International Conference on Power System Technology*, pages 1–7, 2010.

[189] Ouadie Bennouna, Nicolas Heraud, and Zbigniew Leonowicz. Condition monitoring & fault diagnosis system for offshore wind turbines. In *2012 11th International Conference on Environment and Electrical Engineering*, pages 13–17, 2012.

[190] Huageng Luo, Charles Hatch, Matthew Kalb, Jesse Hanna, Adam Weiss, and Shawn Sheng. Effective and accurate approaches for wind turbine gearbox condition monitoring. *Wind Energy*, 17, 05 2014.

[191] Himani Garg and Ratna Dahiya. Modelling and development of wind turbine emulator for the condition monitoring of wind turbine. *International Journal Of Renewable Energy Research*, 5(2):591 – 597, 2015.

[192] Raed Ibrahim and Simon Watson. Advanced algorithms for wind turbine condition monitoring and fault diagnosis. In *WindEurope Summit 2016*, 11 2016.

[193] Michael R. Wilkinson, Fabio Spinato, and Peter J. Tavner. Condition monitoring of generators & other subassemblies in wind turbine drive trains. In *2007 IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives*, pages 388–392, 2007.

[194] Asier Gonzalez-Gonzalez, Alberto Jimenez Cortadi, Diego Galar, and Lorenzo Ciani. Condition monitoring of wind turbine pitch controller: A maintenance approach. *Measurement*, 123:80–93, 2018.

[195] Shahin Hedayati Kia, Humberto Henao, and Gérard-André Capolino. Development of a test bench dedicated to condition monitoring of wind turbines. In *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, pages 2044–2049, 2014.

[196] Marco Raciti Castelli, Alessandro Englaro, and Ernesto Benini. The darrieus wind turbine: Proposal for a new performance prediction model based on cfd. *Energy*, 36(8):4919–4934, 2011.

[197] Maki K. Habib, Samuel A. Ayankoso, and Fusaomi Nagata. Data-driven modeling: Concept, techniques, challenges and a case study. In *2021 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1000–1007, 2021.

[198] Andrew Kusiak and Anoop Verma. A data-mining approach to monitoring wind turbines. *IEEE Transactions on Sustainable Energy*, 3(1):150–157, 2012.

[199] Ayse Kavaz and Burak Barutcu. Fault detection of wind turbine sensors using artificial neural networks. *Journal of Sensors*, 2018, 12 2018.

[200] Jun-Hyun Shin, Yun-Seong Lee, and Jin-O Kim. Fault prediction of wind turbine by using the svm method. In *2014 International Conference on Information Science, Electronics and Electrical Engineering*, volume 3, pages 1923–1926, 2014.

[201] Fangzhou Cheng, Liyan Qu, and Wei Qiao. Fault prognosis and remaining useful life prediction of wind turbine gearboxes using current signal analysis. *IEEE Transactions on Sustainable Energy*, 9(1):157–167, 2018.

[202] Jyh-Yih Hsu, Yi-Fu Wang, Kuan-Cheng Lin, Mu-Yen Chen, and Jenneille Hwai-Yuan Hsu. Wind turbine fault diagnosis and predictive maintenance through statistical process control and machine learning. *IEEE Access*, 8:23427–23439, 2020.

[203] Yueqi Wu and Xiandong Ma. A hybrid lstm-kld approach to condition monitoring of operational wind turbines. *Renewable Energy*, 181:554–566, 2022.

[204] Xiangwu Yan, Yongsheng Jin, Yun Xu, and Ruojin Li. Wind turbine generator fault detection based on multi-layer neural network and random forest algorithm. In *2019 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, pages 4132–4136, 2019.

[205] Wenxian Yang, Richard Court, and Jiesheng Jiang. Wind turbine condition monitoring by the approach of scada data analysis. *Renewable Energy*, 53:365–376, 2013.

[206] Khaoula Tidriri, Nizar Chatti, Sylvain Verron, and Teodor Tiplica. Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges. *Annual Reviews in Control*, 42:63–81, 2016.

[207] Abiodun Olaoye. Wind energy analytics toolbox: Iterative power curve filter. https://towardsdatascience.com/wind-energy-analytics-toolbox-iterative-power-curve-filter-fec258fdb997, 2021. [Online]. Accessed on Apr. 30, 2024.

[208] Annalisa Santolamazza, Daniele Dadi, and Vito Introna. A data-mining approach for wind turbine fault detection based on scada data analysis using artificial neural networks. *Energies*, 14:1845, 03 2021.

[209] Microsoft. The qubit in quantum computing. Available:https://docs.microsoft.com/en-us/azure/quantum/concepts-the-qubit. [Online]. Accessed on Mar. 31, 2024.

[210] Microsoft. Dirac notation. Available:https://docs.microsoft.com/en-us/azure/quantum/concepts-dirac-notation. [Online]. Accessed on Mar. 31, 2024.

[211] Felix Bloch. Nuclear induction. *Phys. Rev.*, 70:460–474, 10 1946.

[212] Fortunato Tito Arecchi, Eric Courtens, Robert Gilmore, and Harry Thomas. Atomic coherent states in quantum optics. *Phys. Rev. A*, 6:2211–2237, Dec 1972.

[213] IBM. Operations glossary. Available:https://quantum-computing.ibm.com/composer/docs/iqx/operations_glossary. [Online]. Accessed on Mar. 31, 2024.

[214] Peter Williston Shor. Quantum computing. *Documenta Mathematica Extra Volume ICM*, 1:467–486, 1998.

[215] Marco Chiani and Lorenzo Valentini. Short codes for quantum channels with one prevalent pauli error type. *IEEE Journal on Selected Areas in Information Theory*, 1(2):480–486, 8 2020.

[216] Adrian Flitney and Derek Abbott. An introduction to quantum game theory. *Fluctuation and Noise Letters*, 2:R175–R188, 12 2002.

[217] Steven E. Landsburg. Quantum game theory. *Wiley Encyclopedia of Operations Research and Management Science*, 2011.

[218] Michal Gordon and Goren Gordon. Quantum computer games: Quantum minesweeper. *Physics Education*, 45:372, 06 2010.

[219] Zeki C. Seskir, Piotr Migdał, Carrie Weidner, Aditya Anupam, Nicky Case, Noah Davis, Chiara Decaroli, İlke Ercan, Caterina Foti, Paweł Gora, Klementyna Jankiewicz, Brian R. La Cour, Jorge Yago Malo, Sabrina Maniscalco, Azad Naeemi, Laurentiu Nita, Nassim Parvin, Fabio Scafirimuto, Jacob F. Sherson, Elif Surer, James R. Wootton, Lia Yeh, Olga Zabello, and Marilù Chiofalo. Quantum games and interactive tools for quantum technologies outreach and education. *Optical Engineering*, 61(8), 2022.

[220] Laura Piispanen, Marcel Pfaffhauser, James R. Wootton, Julian Togelius, and Annakaisa Kultima. Defining quantum games, 2024.

[221] James Wootton. Introducing the world's first game for a quantum computer. https://decodoku.medium.com/introducing-the-worlds-first-game-for-a-quantum-computer-50640e3c22e4. [Online]. Accessed on Mar. 31, 2024.

[222] James Wootton. Quantum battleships: The first game for a quantum computer. https://decodoku.medium.com/quantum-battleships-the-first-multiplayer-game-for-a-quantum-computer-e4d600ccb3f3. [Online]. Accessed on Mar. 31, 2024.

[223] Selim Akl. On the importance of being quantum. *Parallel Processing Letters*, 20:275–286, 09 2010.

[224] Christopher Cantwell. Quantum chess: Developing a mathematical framework and design methodology for creating quantum games, 2019.

[225] Amit Anand, Bikash K. Behera, and Prasanta K. Panigrahi. Solving diner's dilemma game, circuit implementation and verification on the ibm quantum simulator, 2020.

[226] Kory Becker. Flying unicorn: Developing a game for a quantum computer, 2019.

[227] André Ranchin. Quantum go, 2016.

[228] Lu-Feng Qiao, Jun Gao, Zhi-Qiang Jiao, Zhe-Yong Zhang, Zhu Cao, Ruo-Jing Ren, Chao-Ni Zhang, Cheng-Qiu Hu, Xiao-Yun Xu, Hao Tang, Zhi-Hao Ma, and Xian-Min Jin. Quantum go machine, 2020.

[229] Adrian P. Flitney and Derek Abbott. Quantum version of the monty hall problem. *Physical Review A*, 65(6), 6 2002.

[230] Luis Fernando Quezada and Shi-Hai Dong. Quantum version of a generalized monty hall game and its possible applications to quantum secure communications. *Annalen der Physik*, 533(1), 9 2020.

[231] Franz G. Fuchs, Vemund Falch, and Christian Johnsen. Quantum poker—a game for quantum computers suitable for benchmarking error mitigation techniques on nisq devices. *The European Physical Journal Plus*, 135(4), 4 2020.

[232] Ankur Pal, Sanghita Chandra, Vardaan Mongia, Bikash Behera, and Prasanta Panigrahi. Solving sudoku game using a hybrid classical-quantum algorithm. *Europhysics Letters*, 128(4), 2019.

[233] Trevor Glasgow, Oliver Levy, Henry Hilton, and Rafael Brantley. Quantum tetris. Available:https://github.com/dartmouth-cs98/Quantum-Tetris. [Online]. Accessed on Mar. 31, 2024.

[234] Weijing Xiao, Emilio Lari, Billy Ho, Salil Parekh, and Olivier Brückner. Quantum tetris. Available:https://olivierbrcknr.github.io/quantum-tetris/. [Online]. Accessed on Mar. 31, 2024.

[235] Allan Goff, Dale Lehmann, and Joel Siegel. Quantum tic-tac-toe, spooky-coins & magic-envelopes, as metaphors for relativistic quantum physics. *38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, 6 2002.

[236] Jia Ning Leaw and Siew Ann Cheong. Strategic insights from playing quantum tic-tac-toe. *Journal of Physics A: Mathematical and Theoretical*, 43(45):455304, 10 2010.

[237] Qiskit: An open-source framework for quantum computing. Available:https://www.ibm.com/quantum/qiskit, 2019. [Online]. Accessed on Mar. 31, 2024.

[238] Georg Ebers. *Egypt: Descriptive, historical, and picturesque*. Cassell & Co., London, England, 1878.

[239] Edward Falkener. *Games Ancient and Oriental, and How to Play Them: Being the Games of the Greek, the Ludus Latrunculorum of the Romans and the Oriental Games of Chess, Draughts, Backgammon and Magic Squares*. Longmans, Green & Co., London, England, 1892.

[240] Paul F. Perdrizet. The game of morra. *The Journal of Hellenic Studies*, 18:129–13, 1898.

[241] Charlton T. Lewis and Charles Short. mico. In *A Latin Dictionary*. Clarendon Press, Oxford, England, 1879.

[242] Marcus Tullius Cicero. *De Officiis. With An English Translation by Walter Miller*. William Heinemann Ltd., London, England, 1913.

[243] Hans Ulrich Vogel and Günter Dux. *Concepts of Nature: A Chinese-European Cross-Cultural Perspective*. Brill, Netherlands, 2010.

[244] Yabla. cāi quán. https://chinese.yabla.com/chinese-english-pinyin-dictionary.php?define=+%E7%8C%9C%E6%8B%B3+. [Online]. Accessed on Mar. 31, 2024.

[245] Yabla. huá quán. https://chinese.yabla.com/chinese-english-pinyin-dictionary.php?define=+%E5%88%92%E6%8B%B3+. [Online]. Accessed on Mar. 31, 2024.

[246] Yabla. mǔ zhàn. https://chinese.yabla.com/chinese-english-pinyin-dictionary.php?define=+%E6%8B%87%E6%88%98+. [Online]. Accessed on Mar. 31, 2024.

[247] Xie Zhaozhe. *A Fivefold Miscellany (Wu Za Zu)*. 1608.

[248] Franco Delogu, Madison Barnewold, Carla Meloni, Enrico Toffalini, Antonello Zizi, and Rachele Fanari. The morra game as a naturalistic test bed for investigating automatic and voluntary processes in random sequence generation. *Frontiers in Psychology*, 11, 2020.

[249] Davide Onida. Sa murra game. http://samurra.it/. [Online]. Accessed on Mar. 31, 2024.

[250] AGI editorial staff. Nessuno batte i sardi a morra, nemmeno un robot. https://www.agi.it/cronaca/news/2021-07-25/sardegna-robot-perde-contro-umani-gioco-morra-13377028/. [Online]. Accessed on Mar. 31, 2024.

[251] IBM. What is quantum computing? https://www.ibm.com/quantum-computing/what-is-quantum-computing/. [Online]. Accessed on Mar. 31, 2024.

[252] Daniela Frauchiger, Renato Renner, and Matthias Troyer. True randomness from realistic quantum devices, 2013.

[253] Antonio Costantino Marceddu and Bartolomeo Montrucchio. A quantum adaptation for the morra game and some of its variants. *IEEE Transactions on Games*, pages 1–9, 2023.

[254] Vlatko Vedral, Adriano Barenco, and Artur Ekert. Quantum networks for elementary arithmetic operations. *Physical Review A*, 54(1):147–153, 7 1996.

[255] Steven A. Cuccaro, Thomas G. Draper, Samuel A. Kutin, and David Petrie Moulton. A new quantum ripple-carry addition circuit, 2004.

[256] Vlatko Vedral, Adriano Barenco, and Artur Ekert. Improved quantum ripple-carry addition circuit. *Science China Information Sciences*, 59(042406), 2016.

[257] Thomas G. Draper, Samuel A. Kutin, Eric M. Rains, and Krysta M. Svore. A logarithmic-depth quantum carry-lookahead adder, 2004.

[258] Agung Trisetyarso and Rodney Van Meter. Circuit design for a measurement-based quantum carry-lookahead adder. *International Journal of Quantum Information*, 08(05):843–867, 8 2010.

[259] Himanshu Thapliyal, Edgard Muñoz-Coreas, and Vladislav Khalus. T-count and qubit optimized quantum circuit designs of carry lookahead adder, 2020.

[260] Phil Gossett. Quantum carry-save arithmetic, 1998.

[261] Lorenzo Lippi. *Malmantile racquistato*. S.A.S alla Condotta, Florence, Italy, 1688.

[262] William Smith, William Wayte, and George Eden Marindin. *A Dictionary of Greek and Roman Antiquitie*. John Murray, London, England, 1890.

[263] Hugo Von Trimberg. *Der Renner*. Litterarischer Verein, Stuttgart, Germany, 1908.

[264] Antonio Costantino Marceddu, Nicola Dilillo, Marco Russo, Renato Ferrero, and Bartolomeo Montrucchio. A quantum adaptation to roll truly random dice in role playing games. In *2023 IEEE Conference on Games (CoG)*, pages 1–4, 2023.

[265] William J. White, Jonne Arjoranta, Michael Hitchens, Jon Peterson, Evan Torner, and Jonathan Walton. Tabletop role-playing games. In *Role-Playing Game Studies*, pages 63–86. Routledge, 2018.

[266] Solrun Brenk Rønning and Stål Bjørkly. The use of clinical role-play and reflection in learning therapeutic communication skills in mental health education: an integrative review. *Advances in medical education and practice*, pages 415–425, 2019.

[267] Mark Matthews, Geri Gay, and Gavin Doherty. Taking part: role-play in the design of therapeutic systems. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 643–652, 2014.

[268] Daniel Luccas Arenas, Anna Viduani, and Renata Brasil Araujo. Therapeutic use of role-playing game (RPG) in mental health: A scoping review. *Simulation & Gaming*, 53(3):285–311, 2022.

[269] James R. Wallace, Joseph Pape, Yu-Ling Betty Chang, Phillip J. McClelland, T. C. Nicholas Graham, Stacey D. Scott, and Mark Hancock. Exploring automation in digital tabletop board game. In *Proceedings of the ACM 2012 conference on computer supported cooperative work companion*, pages 231–234, 2012.

[270] Alberto Manoel dos Santos Silva Junior and Victor Travassos Sarinho. TaRCoS: Providing a combat simulation tool for tabletop role-playing games. In *20th Brazilian Symposium on Computer Games and Digital Entertainment*, pages 344–348. SBC, 2021.

[271] Peter Kietzmann, Thomas C Schmidt, and Matthias Wählisch. A guideline on pseudorandom number generation (PRNG) in the IoT. *ACM Computing Surveys (CSUR)*, 54(6):1–38, 2021.

[272] Kamalika Bhattacharjee and Sukanta Das. A search for good pseudo-random number generators: Survey and empirical studies. *Computer Science Review*, 45:100471, 2022.

[273] Alan M. Frieze, Ravindran Kannan, and Jeffrey C. Lagarias. Linear congruential generators do not produce random sequences. In *25th Annual Symposium on Foundations of Computer Science, 1984.*, pages 480–484. IEEE, 1984.

[274] Miguel Herrero-Collantes and Juan Carlos Garcia-Escartin. Quantum random number generators. *Reviews of Modern Physics*, 89(1):015004, 2017.

[275] Vaisakh Mannalath, Sandeep Mishra, and Anirban Pathak. A comprehensive review of quantum random number generators: Concepts, classification and the origin of randomness. *arXiv preprint arXiv:2203.00261*, 2022.

[276] Anish Saini, Athanasios Tsokanos, and Raimund Kirner. Quantum randomness in cryptography—a survey of cryptosystems, RNG-based ciphers, and QRNGs. *Information*, 13(8):358, 2022.

[277] Nicola Dilillo. Interactive 3D Quantum Dice Roller. https://github.com/CARDIGANSPoliTo/quantum-dice-roller. [Online]. Accessed on Mar. 31, 2024.

[278] Anton Natarov. Online 3D dice roller. http://a.teall.info/dice/. [Online]. Accessed on Mar. 31, 2024.

[279] Matteas Eden. Interactive 3D dice roller. https://github.com/Matteas-Eden/dice-roller. [Online]. Accessed on Mar. 31, 2024.

Appendix A

Games for Quantum Computing

During my Ph.D., I also worked on projects related to building game adaptations using quantum computing. In particular, I worked on a quantum adaptation of the Morra game and Role-Playing Game (RPG) dice. They will be discussed in the following Sections.

A.1 Brief Introduction to Quantum Computing

In quantum computing, the *Quantum Binary Digit (qubit)* is the fundamental unit for information processing. Each state in which the qubit can be found can be described as a superposition of two orthonormal basis states, analogous to the states of classical Binary Digits (bits) [209]. These states, represented as $|0\rangle$ and $|1\rangle$, are commonly represented using *Dirac notation*, also known as *bra-ket notation* [210], and can be visualized geometrically using the *Bloch sphere* [211, 212].
Unlike classical computational gates, all gates used in quantum computing must be reversible to ensure that the original information can always be reconstructed. For example, in classical arithmetic, a subtraction such as 8-5=3 leads to a loss of information since it is impossible to recover the operands exclusively from the result. This loss cannot happen in quantum computing since such irreversible operations are not allowed.
A *quantum circuit* is a theoretical model used in quantum information theory to perform quantum computations analogous to classical circuits. Quantum circuits are commonly used in *gate-based* quantum computing, where a computation consists of

a sequence of quantum gates, measurements, initializations of qubits to known values, and potentially other actions. Here are the descriptions of some gates typically used in gate-based quantum computing [213]:

- *NOT gate*: reverses the state of the qubit from $|0\rangle$ to $|1\rangle$ and vice versa. It is graphically depicted as a rounded block with a + or a square block with an X.
- *CNOT gate* (*Controlled-NOT*): reverses the state of a target qubit if the control qubit is set to $|1\rangle$. It is graphically represented as a rounded block with a + near the target qubit and a line terminating in a circle for the control qubit.
- *Toffoli or CCNOT gate* (Controlled-Controlled-NOT): reverses the state of a target qubit if both control qubits are set to $|1\rangle$. Can implement NOT, AND, and Exclusive OR (XOR) operations. It is graphically shown with a + near the target qubit and a line with two circles for the control qubits.
- *Hadamard gate*: sets the state of a qubit in an equal superposition of $|0\rangle$ and $|1\rangle$, achieving true randomness. It is graphically depicted as a square block with an H.
- *SWAP gate*: swaps the state of two qubits. It is graphically represented as a line with Xs at the ends.
- *RX gate*: rotates the qubit state around the x-axis by a desired angle. It is graphically shown as a square block with RX and possibly the rotation value.
- *RY gate*: rotates the qubit state around the y-axis by a desired angle. It has a similar graphical depiction to the previous gate.
- *RZ gate*: rotates the qubit state around the z-axis by a desired angle. Even this gate has a graphical representation similar to the previous ones.

Quantum computers are not flawless machines. For optimal operation, they should be completely shielded from the surrounding environment. However, even introducing an input into a quantum computer breaks this shielding and produces inevitable perturbations. Quantum error correction and mitigation algorithms can improve the results obtained from calculations. However, much research is still needed to reduce the effects of noise and decoherence further [214, 215].

A.2 Quantum Games

When searching for the term *quantum games*, it is expected to find results concerning quantum game theory, which is simply the extension of game theory to the quantum world [216, 217]. This is beyond the scope of the subject of this Section, which focuses instead on the very concept of gaming as applied to quantum computers. Until now, few attempts have been made to create a complete definition of quantum games. Quantum games, as per [218], are *computer games where the rules of the game are based on quantum principles, such as superposition, entanglement, and the collapse of the wave function*. A broader definition describes quantum games as *computer (or video) games with one or more of the phenomena from quantum physics embedded in their game mechanics* [219]. The problem with these definitions is that they only encompass educational aspects, leaving out other equally important ones. A recent article [220] has introduced a Three Dimensional (3D) framework for distinguishing quantum games, comprising the following aspects:

- *Quantum physics*: incorporate quantum physics concepts into graphics, mechanics, or other game elements.
- *Quantum technologies*: examine the impact of quantum computers or software on development and gameplay.
- *Scientific purposes*: testing the capabilities and limits of quantum computers.

According to these dimensions, quantum games can be defined as *games that reference the theory of quantum physics, quantum technologies, or quantum computing through perceivable means, connect to quantum physics through a scientific purpose, or use quantum technologies*. This definition fully captures the continually evolving nature of the field and is currently the most valued one.
Several quantum games have been developed so far. Some examples are proposed below:

- *Cat/Box/Scissors*: the first quantum game capable of running on a quantum computer. It is a variation of Rock/Paper/Scissors with an arbiter and four potential opponents to play with [221].
- *Quantum Battleship*: the first turn-based multiplayer game that can run on a quantum computer. It is essentially a simplified version of Battleship where

one player places a boat, and the other has to guess where it has been placed [222].

- *Quantum Chess*: an adaptation of classical chess for quantum computers, implementations proposed so far that focus on teaching the properties of quantum mechanics [223, 224].
- *Diner's Dilemma*: a classical problem in game theory and economics, solved in its four-player variant using quantum circuits [225].
- *Flying Unicorn*: a game in which a unicorn must try to fly over a castle and in which the uncertainty, superposition, and qubit measurement properties are explored [226].
- *Quantum Go*: so far, two implementations of the classic game of Go have been created to make quantum mechanics accessible and serve as a test platform for new AI algorithms [227, 228].
- *Quantum Minesweeper*: a variation of the classic computer game aiming to teach basic quantum mechanics [218].
- *Monty Hall problem*: various quantum versions of this probability puzzle were proposed over time, with potential applications in quantum information [229, 230].
- *Quantum Poker*: a variant of Texas Hold 'em for teaching quantum computing phenomena [231].
- *Quantum Sudoku*: quantum circuits capable of solving small 4x4 Sudoku grids (generally termed as Shi Doku), integrating duality computing and probabilistic approaches [232].
- *Quantum Tetris*: educational versions of the popular game focus on teaching general concepts of quantum theory and visualizing the effects of quantum noise on the game itself [233, 234].
- *Quantum Tic Tac Toe*: two versions have been developed since. The former serves as a didactic metaphor for nonlocality and quantum physics [235], while the latter achieves a minimal quantization of the game itself [236].

A.3 Tools

The study and implementation of circuits were facilitated using IBM Qiskit, a prominent open-source Software Development Kit (SDK) tailored for quantum computing solutions [237]. In Qiskit and the quantum computing field, each instance of circuit execution is called *shot*; this term will also be used in subsequent Sections.

A.4 Quantum Morra

The *Morra* game is rather ancient. The first evidence of the game dates back to the iconography in the tombs of Beni Hasan, Egypt, around 2000 Before the Common Era (BCE) [238, 239]. Subsequent traces appear in Greek vase paintings, one of which depicts Helen of Troy and Paris [240], and in documents from the Roman Empire which call it *micatio* or *micare digitis* [241]. Cicero even mentions it in his *De officiis*, indicating the potential popularity facilitated by the vast reach of the Roman Empire [242]. In China, it was called *cāi quán*, *huá quán*, or *mǔ zhàn*, and these names are still used today [243–246]. The game still enjoys good popularity today, being played in countries such as Italy, France, Greece, Spain, China and Mongolia.

Morra has some pretty simple rules. Two players simultaneously use one hand to mime a number from 1 to 5 and shout another number between 2 and 10, corresponding to the presumed sum of the mimed numbers. A point is awarded to one of the players if they correctly guesses the sum; however, no points will be awarded if both players guess correctly. The threshold of points needed to win is variable: for example, in the Sardinian version, called *sa Murra*, they are 16. In the standard version of the game, closed fists are equivalent to one; for this reason, the minimum number that can be obtained is two. At the same time, ten is nicknamed *Morra* or *all*. Due to the similarities between the game modes, rock-paper-scissors is also known as Chinese Morra, although it is to be considered a different and separate game.

Despite its simplicity, Morra represents a challenge and requires good cooperation of perceptual, motor, and cognitive skills [247, 248]. At a professional level, the game has quite a high pace and can stick to the round per second. For this reason, professional players have to train a lot. Specifically, they must be able to discern their opponents' patterns and, at the same time, try to remain unpredictable.

Morra has seen adaptations for PCs and smartphones that allow to play against Artificial Intelligence (AI) or other players in online multiplayer [249]. Several robots capable of playing against humans have also been developed, such as Gavina2121, who competed in Bitti (located in Sardinia, Italy) against humans and suffered a resounding defeat [250].

A.4.1 Implementing a Quantum Random Player

The high speed of processing complex calculations and true random number generation capabilities of quantum computers could revolutionize Morra's gameplay, making predictions impossible even for expert players [251, 252]. A quantum random player who can play against a real player should be able to provide a random number between 1 and 5, representing the indicated number, and another number between 2 and 10 equal to the guessed sum. The development of a circuit able to generate two random numbers between 1 and 5 and a further one to add the two previously generated numbers is therefore necessary. This is exactly what has been realized so far, to which the creation of a quantum arbiter capable of verifying who won between the actual player and the quantum one must be added [253].

A.4.1.1 Generation of Equally Probable Numbers Between 1 and 5

The initial step in creating a quantum random player for the Morra game involves making a circuit representing the number the human hand mimics. Base two is the fundamental reference in quantum computers and classical ones. With n qubits, it becomes possible to represent up to 2^n distinct values or states. Therefore, creating a true randomizer circuit for powers of two numbers is relatively simple since it requires, for each bit, the insertion of a Hadamard gate and a measurement operation. Complexity arises when generating numbers within different ranges. Reducing the probability of obtaining certain states to zero is imperative in these cases. Extracting a number between 1 and 5 with equal probability is necessary to implement a quantum version of the Morra game. Since five is not a power of two, it is necessary to consider the next power of two greater than that number, which is eight, and eliminate the likelihood of reaching three states.
A potential circuit capable of achieving this is illustrated in Figure A.1. It includes:

- An RY gate, which rotates the q_2 qubit by $\frac{\pi}{1.41877626883}$ radians.
- A Toffoli gate, setting q_1 to 1 only if q_0 and q_2 are both 1.
- Two Hadamard gates, with the first placing q_0 in equal superposition while the second resets its value to 0 with a probability of 100%. The superposition of q_0 facilitates the triggering of the Toffoli gate.
- A NOT gate, which inverts the value of q_2.
- Three measurement operations, necessary to obtain the final value of each qubit.

This circuit was created using Monte Carlo techniques. Therefore, numerous tests were conducted to find a circuit capable of canceling three of the eight output states. Once found, it was optimized to minimize gate counts and rough output states. Next, a brute force method was employed to determine the precise RY rotation to ensure an equal probability (20%) for each state. The resulting circuit can generate numbers between 0 and 4 and will subsequently be called *five-values number generator circuit*. One of the possible methods to improve the results obtained via software is to perform measurement calibration. This involves preparing 2^n basis input states and evaluating the probability of obtaining counts by measuring in alternative basis states. Using IBM simulators, such as QASM or Aer, it is possible to calculate the noise-free calibration matrix, which, in this case, is a simple identity matrix. Applying the noise model of a real quantum computer on simulators allows the calibration matrix for that quantum computer to be calculated to mitigate measurement errors. Figure A.2 reports the result of running the five-valued number generator circuit for 16384 shots using the IBM QASM simulator. It was executed two times. In the first execution, there was no noise (Noiseless), while in the second execution, the noise model of

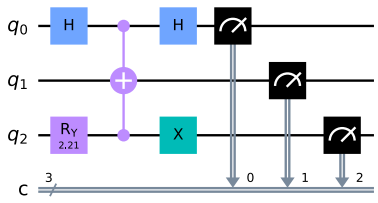

FIGURE A.1: The five-values number generator circuit. This image was taken from [253].

IBM Quito, a real five qubit quantum computer, was applied (Noisy). It is clear that, unlike the optimal results obtained by the IBM QASM simulator in the absence of noise, eight values are returned instead of the expected five. Figure A.2 also shows that, by applying measurement calibration, it is possible to improve the results obtained (Mitigated).

A.4.1.2 Implementation of a Quantum Adder

The second step in creating a quantum random player for the Morra game involves creating a circuit to calculate the guessed sum. The most straightforward implementation of a quantum and non-quantum adder is to use a *half adder*. It can take two numbers, A and B, and produce their sum and carry. However, it does not consider any carryovers that may arise from a previous sum. To add more than two bits, it is, therefore, necessary to use a *full adder*, which can also consider any carryover of the previous addition. *Multi-bit adders* can be obtained by linking multiple full adders. Different multi-bit adders exist depending on how the carry is handled, and many are available in quantum computing:

- *Quantum Ripple-Carry Adder (QRCA)*: basic adder that calculates the carry for each bit and uses it in subsequent addition [254–256].

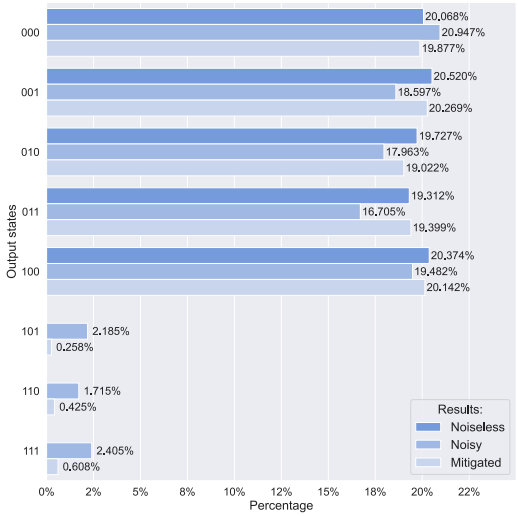

FIGURE A.2: The Noiseless, Noisy, and Mitigated outcomes acquired from executing the five-values number generator circuit for 16384 shots on the IBM QASM Simulator. This image was taken from [253].

- *Quantum Carry-Lookahead Adder (QCLA)*: improves QRCA by eliminating carry propagation delay but is more complex. It is currently the fastest quantum adder known, especially for many-bit operations [257–259].
- *Quantum Carry-Save Adder (QCSA)*: calculates subtotals and carries for multiple inputs, then combines them for the final result [260].

As seen previously, in the Morra game, the numbers to be added can be represented in three bits. Therefore, a simple multi-bit adder, illustrated in Figure A.3, will be used. Since existing quantum computers have limited available qubits, a quantum circuit comprising a half adder for the first qubit and two full adders for the remaining ones was used, lowering the number of qubits needed for the sum to nine. Given two three-bit numbers, A and B, starting from the Most Significant Bit (MSB), the bits should be inserted into the circuit as follows:

$$A[2] \rightarrow q_6,\ A[1] \rightarrow q_3,\ A[0] \rightarrow q_0$$

$$B[2] \rightarrow q_7,\ B[1] \rightarrow q_4,\ B[0] \rightarrow q_1$$

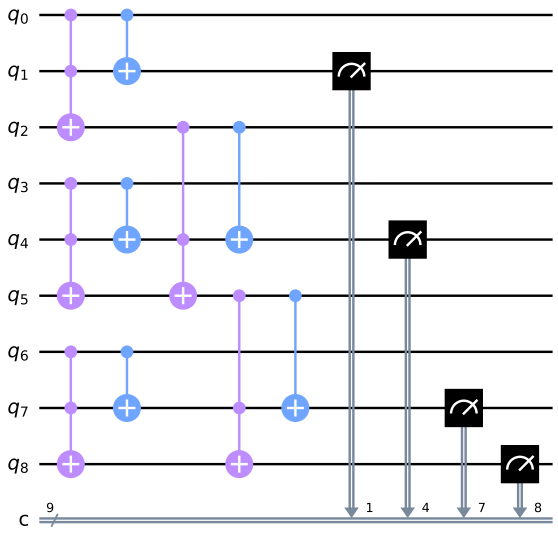

FIGURE A.3: The adder circuit designed for the Morra game allows the computation of the sum of two numbers represented with three qubits. This image was taken from [253].

After the calculation, the highest indexed output bits will contain the MSBs of the resulting sum, so it will read as follows:

$$O[3] \to q_8,\ O[2] \to q_7,\ O[1] \to q_4,\ O[0] \to q_1$$

Since the sum of the two numbers guessed by the players is between 2 and 10, this value cannot be contained in three bits; four bits are therefore needed.

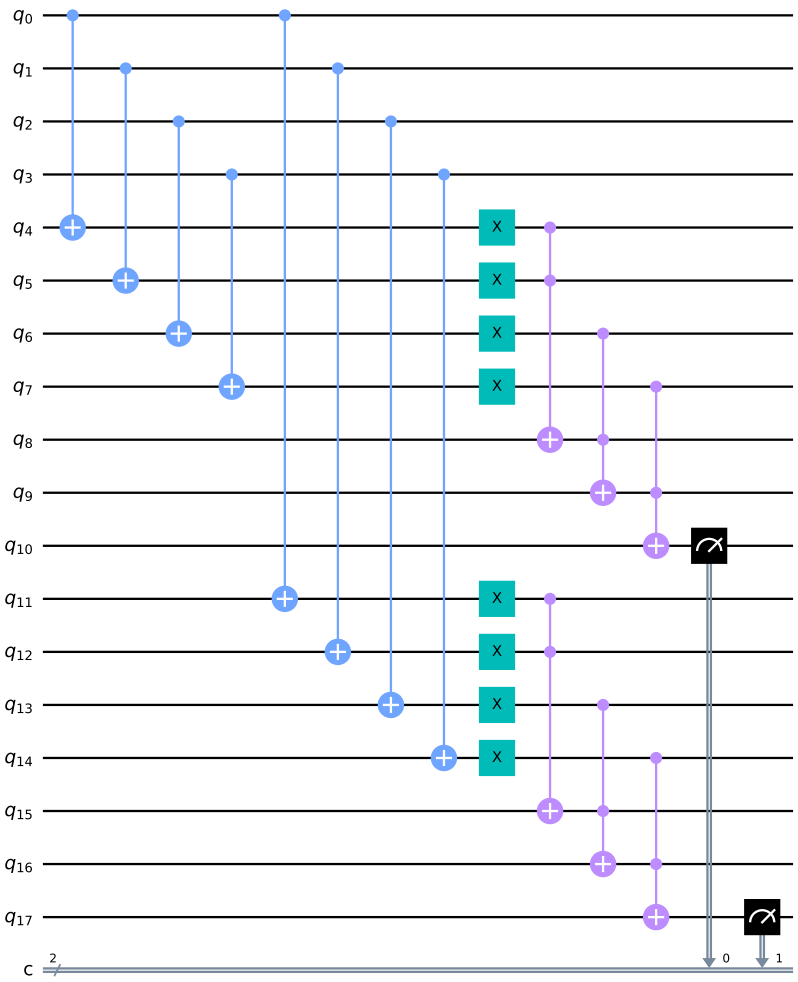

FIGURE A.4: A screenshot depicting the arbiter circuit, which compares the players' guessed sum with the effective one. This image was taken from [253].

A.4.2 Implementation of a Quantum Arbiter

The completion of the quantum version of the Morra game is possible by introducing a circuit that acts as an *arbiter* and compares the two sums guessed by the players with the actual sum. Each game round can have three possible scenarios:

- Both sums are equal to the actual one.
- Both sums differ from the actual one.
- One of the sums is equal to the actual one.

The comparison between two qubits can be done by using a CNOT gate and measuring the target qubit: if the qubits have the same value, the output is 0; otherwise, it is 1. Extending this principle, it is possible to use twelve qubits to compare the four qubits of control of the actual sum (C) with the respective eight target qubits of

```
QUANTUM MORRA
Welcome to the Quantum Morra game!
The game will end once 16 points have been reached

Please enter 5 if you want to play with 5 numbers (vanilla version)
or 6 if you to play with 6 numbers (allowing zero variant): 5

ROUND 1
Digit the hand mimicked number: 5
Digit the supposed sum: 7
Calculation of the Quantum Player number...
Job Status: job has successfully run
QUANTUM PLAYER NUMBER: 2
Calculation of the Quantum Player supposed sum...
Job Status: job has successfully run
QUANTUM PLAYER SUPPOSED SUM: 5
Addition and comparison of the supposed sums...
Job Status: job has successfully run
COMPARISONS: 01

Player supposed the correct sum!
Points - Player: 1 , Quantum Player: 0


ROUND 2
Digit the hand mimicked number:
```

FIGURE A.5: A screenshot from the Quantum Morra game. At the beginning of the game, the user can choose whether to allow zeros or not. This image was taken from [253].

the guessed sums (S_1, S_2). Starting from their MSBs, the sums are entered into the circuit as follows:

$$C[3] \rightarrow q_0,\ C[2] \rightarrow q_1,\ C[1] \rightarrow q_2,\ C[0] \rightarrow q_3$$

$$S_1[3] \rightarrow q_4,\ S_1[2] \rightarrow q_5,\ S_1[1] \rightarrow q_6,\ S_1[0] \rightarrow q_7$$

$$S_2[3] \rightarrow q_{14},\ S_2[2] \rightarrow q_{13},\ S_2[1] \rightarrow q_{12},\ S_2[0] \rightarrow q_{11}$$

It is then possible to apply a downsampling mechanism to measure the output of two single qubits, sufficient to represent the abovementioned scenarios. First, the eight target qubits values are inverted using NOT gates, which are intended to change the values of 0 back to 1 following the previous operation. Secondly, six Toffoli gates are used to downsample the eight qubits to two qubits. The output of this downsampling operation will be 1 if the two numbers are equal, i.e., all the qubits that compose them are identical to one; vice versa, it will be 0 if the two numbers are different. The final arbiter circuit, capable of performing the abovementioned operation, is illustrated in Figure A.4.

A.4.3 Assembly of Parts and Analysis of Operation

The final circuit was built by combining the five-values number generator circuit, adder, and arbiter. A text-based interface has also been incorporated to allow users to play against the quantum random player. An image depicting the final game interface is presented in Figure A.5. During each round, the user is asked to enter two numbers: the value mimed with the hand and the guessed sum. Next, the quantum random player performs the same task. After both players have made their selection, the actual sum is calculated, and the arbiter decides the round winner. As

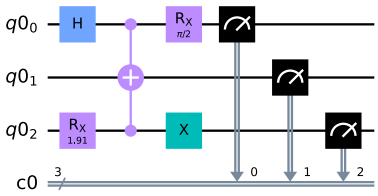

FIGURE A.6: The six-values number generator circuit. This image was taken from [253].

per the Sardinian Morra rules, the default number of points required to win is 16, although it can be changed at will.

A.4.4 Variant with Zero Allowed

Among the different variations available for the Morra game, there is one in which players can also use the number zero. It is mimed simply by leaving the hand closed. This addition involves some changes to the rules of the game:

- Firstly, the range of extractable numbers goes from 1-5 (5 distinct numbers) to 0-5 (6 distinct numbers).
- Secondly, the range of possible sums that players can guess is from 2-10 to 0-10.

While expanding the adding range does not affect the adder circuit, increasing the extractable numbers requires creating a new number generator circuit.
There are various methods for building a circuit capable of extracting a number between 0 and 5 with equal probability. Figure A.6 presents one of these circuits, defined as *six-values number generator circuit*, which was derived from the five-values number generator circuit by implementing the following substitutions:

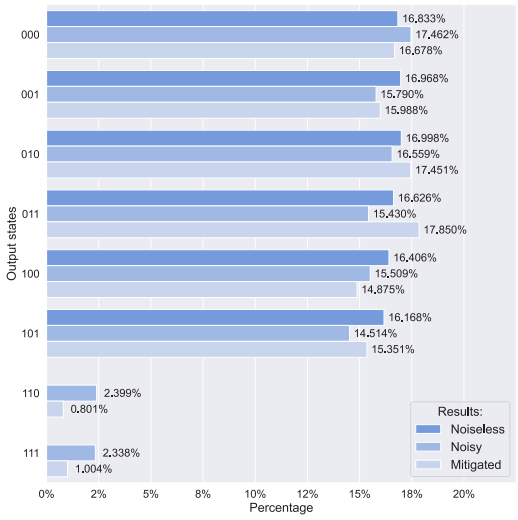

FIGURE A.7: The Noiseless, Noisy, and Mitigated outcomes acquired from executing the six-values number generator circuit for 16384 shots on the IBM QASM Simulator. This image was taken from [253].

- One of the Hadamard gates was replaced with an RX gate, which rotates the q_0 qubit by $\frac{\pi}{2}$.
- The RY gate has been replaced with an RX gate, which rotates the q_2 qubit by $\frac{\pi}{1.644267775}$ radians.

In the previous number generator circuit, the first Hadamard gate acted on qubit q_0, placing it in equal superposition and activating the Toffoli gate if qubit q_2 was set to 1, while the second Hadamard gate reset the value of qubit q_0 to 0 with 100% chance. However, in the new circuit, using a single Hadamard gate followed by an RX gate preserves the superposition state of the q_0 qubit, thus introducing an additional state, as expected. Like the prior circuit, the precise rotation of the RX gate was determined through a brute force approach to obtain a probability of 16.666667% for each output state. As done previously for the five-values number generator circuit, Figure A.7 reports the result of running the six-valued number generator circuit for 16384 shots using the IBM QASM simulator.

```
QUANTUM ODDS AND EVENS
Welcome to the Quantum Odds And Evens game!
The game will end once 3 rounds have been reached

Please enter 5 if you want to play with 5 numbers (vanilla version)
or 6 if you to play with 6 numbers (allowing zero variant): 5
Please enter 0 to assign you the even numbers or 1 to assign you the odd numbers: 1

ROUND 1
Digit the hand mimicked number: 3
Calculation of the Quantum Player number...
Job Status: job has successfully run
QUANTUM PLAYER NUMBER: 5
Calculation of the sum of the previously drawn numbers...
Job Status: job has successfully run
ADDITION: 8

The number is EVEN!
Quantum Player scores one point!


ROUND 2
Digit the hand mimicked number:
```

FIGURE A.8: A screenshot from the Quantum Odds and Evens game. At the beginning of the game, the user can choose whether to allow zeros or not. This image was taken from [253].

A.5 Quantum Odds and Evens

Odds and Evens is a variant of the Morra game known by the ancient Greeks as *artiazein* and by the Romans as *ludere par impar* [261, 262]. A medieval reference dating back to around 1300 BCE is found in Von Trimberg's *Der Renner* [263]. Even today, it continues to be very popular especially among children, and is known by various names, including *choosies*, *pick*, and *bucking up*.
The rules of Odds and Evens are more straightforward than those of Morra. At the start of the game, players agree on who will earn points for odd or even numbers. Next, a synchronization mechanism is used via a shouted count such as *One, two, three, shoot!* or similar, and both players quickly and simultaneously imitate two numbers between 1 and 5, or 0 and 5 if zero is allowed. If the sum of the two mimed numbers is even, the player who takes the even numbers scores a point, and vice versa for the player who takes the odd numbers. Players can conclude the game after a single round, a best of three rounds, or multiple rounds. The realization of a quantum version of the Odds and Evens game does not require additional circuits compared to those previously discussed for the Morra game:

- The quantum random player can be made using the five- or six-values number generator circuits.
- The values chosen by the actual player and the quantum random player can be added using the abovementioned multi-bit quantum adder.
- Furthermore, the presence of an arbiter is not necessary: parity can be verified simply by examining the Least Significant Bit (LSB) of the sum: if it is 0, the number is even, while if it is 1, the number is odd.

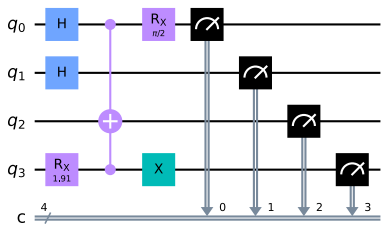

FIGURE A.9: The twelve-values number generator circuit. This image was taken from [264].

An image of the final game, which performs these precise steps, is displayed in Figure A.8.

A.6 Quantum Dice for Role-Playing Game

In a RPG, players create and develop characters with unique abilities, personalities, and backgrounds, engaging in a collaborative narrative [265]. Initially, players gathered around a table, conjuring up a fictional realm through dialogue and exchange, leading to the term Tabletop Role-Playing Games (TRPGs). Other formats later emerged, such as Live-Action Role-Playing (LARP) games and Massively Multi-

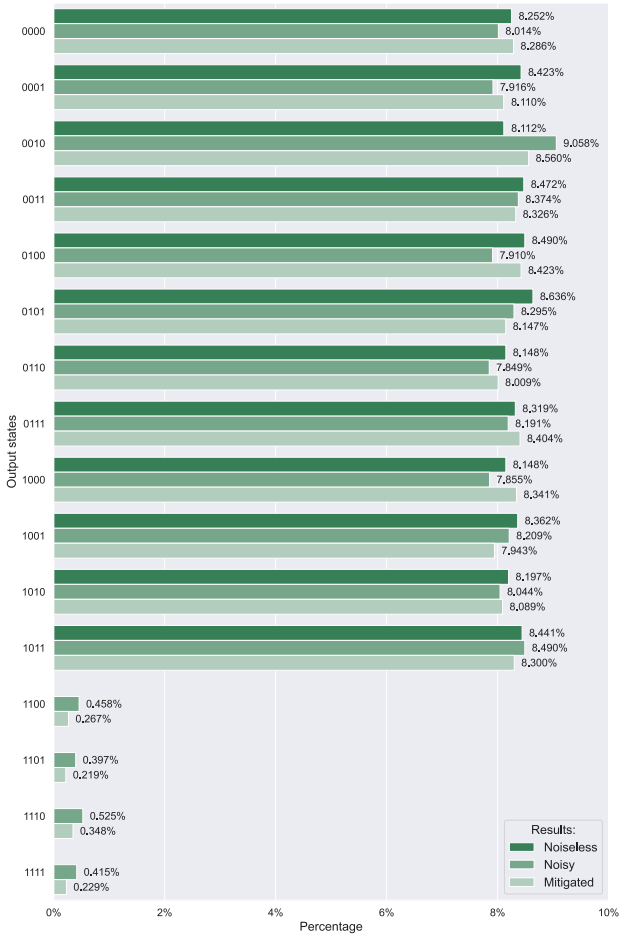

FIGURE A.10: The Noiseless, Noisy, and Mitigated outcomes acquired from executing the twelve-values number generator circuit for 16384 shots on the IBM Aer Simulator. This image was taken from [264].

player Online Role-Playing Games (MMORPGs).
Typically, RPGs involve exploring fictional worlds, dealing with challenges such as combat, and trials that require players to use the abilities of their character to overcome obstacles. The outcome depends partly on the governing system of rules, the player's choices, and the guidance of the Game Master (GM), who acts as judge and narrator. Randomness, often facilitated by dice or cards, plays a fundamental role in determining the success or failure of the player's actions.
In RPGs, dice are used having the shapes of the Platonic solids such as the tetrahedron (4 faces), the cube (6 faces), the octahedron (8 faces), the dodecahedron (12 faces), and the icosahedron (20 faces) with the addition of the pentagonal trapezohedron (10 faces). During the game, they are thrown individually or in groups. For example, a 20-sided die often determines the success of actions such as attacks or spells, while another die determines any resulting damage. At the same time, rolling two 10-sided dice produces a *percentile* roll, yielding results from 1 to 100.
RPGs go beyond entertainment, serving as therapeutic tools for developing social, communication, and behavioral skills [266–268]. In recreational and therapeutic contexts, RPGs can take various forms, with automation enhancing gameplay, alleviating repetitive tasks, and improving accessibility to resources [269, 270]. Random Number Generators (RNGs) support gameplay, but their frequent use and impact on game mechanics underline their vital role.
Traditionally, computer programs implement algorithms to generate random values, with notable examples including the XOR shift, Mersenne Twister, congruential linear generator, and Well Equidistributed Long-period Linear (WELL) [271, 272]. These algorithms, called Pseudo-Random Number Generators (PRNGs), produce pseudorandom values, simulating true randomness through deterministic computation. However, PRNGs have limitations compared to true random numbers, such as the correlation between successive values, bias towards specific values, and period-

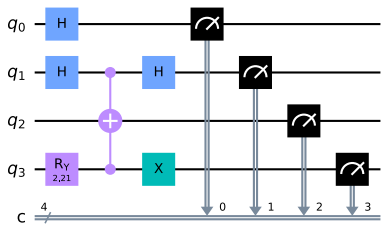

FIGURE A.11: The ten-values number generator circuit. This image was taken from [264].

icity [273]. In this work, the limitations of PRNGs were addressed by leveraging quantum computing to generate truly random numbers [274]. Quantum mechanics, governed by probabilistic laws, offers intrinsic randomness at the subatomic level, allowing the creation of genuinely random numbers. While recent literature discusses various quantum random number generators [275, 276], this Section introduces quantum circuits optimized for generating random numbers within the dice range of RPGs [264].

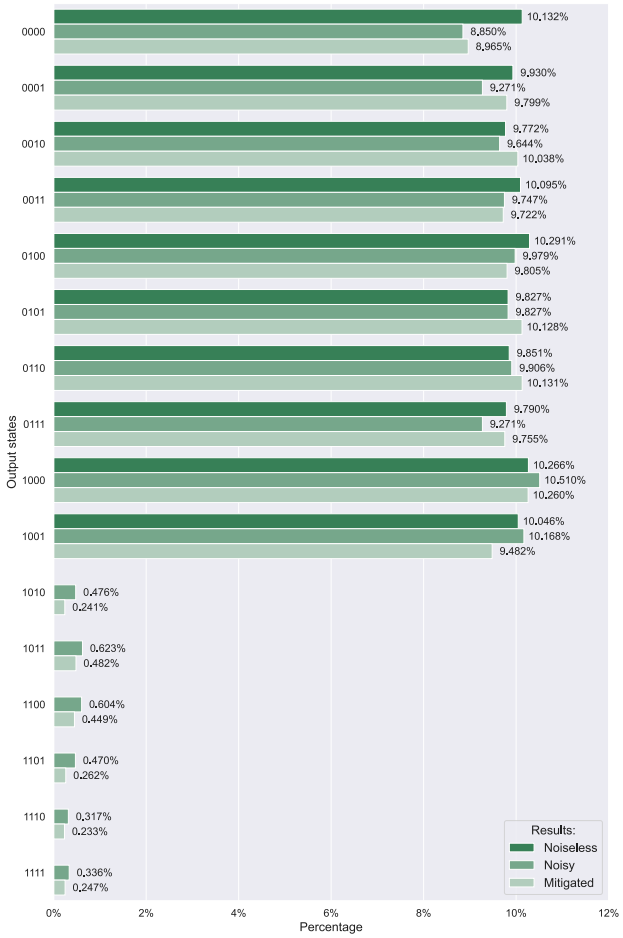

FIGURE A.12: The Noiseless, Noisy, and Mitigated outcomes acquired from executing the ten-values number generator circuit for 16384 shots on the IBM Aer Simulator. This image was taken from [264].

A.6.1 Implementation of 4 and 8-Sided Dice

Implementing a quantum version for RPG dice involves creating multiple circuits, each corresponding to a different die. Creating circuits for 4-sided and 8-sided dice is straightforward, as these numbers are powers of two. They can be obtained simply by applying a Hadamard gate for each qubit. While there are alternative circuits to achieve this, the mentioned approach is among the simplest.

A.6.2 Implementation of 6, 10, 12, and 20-Sided Dice

Implementing circuits for dice with six, ten, twelve, and twenty sides is more complex since these numbers do not have this property. Therefore, additional operations must be performed to reduce the probability of obtaining unwanted states to 0. A circuit capable of generating equally probable numbers between 0 and 5, as discussed for the Morra game, can be repurposed to simulate a six-sided die [253]. Now, it is necessary to find a way to simulate the three remaining dice. Since twelve is a multiple of six, it is possible to obtain a circuit capable of generating twelve equally probable numbers simply by starting from the six-values number generator circuit. Since three qubits are not enough to represent twelve states, an additional qubit must be added and put into superposition through a Hadamard gate. Subsequently, adjustments must be made to the remaining gates to ensure contiguous results within the range of 0-11. The resulting circuit is depicted in Figure A.9, while its execution outcomes are depicted in Figure A.10. Since ten and twenty are multiples of five, it is possible to use the same previous workflow starting from the five-valued number generator circuit, adding qubits, superposing them, and adjusting the result to have

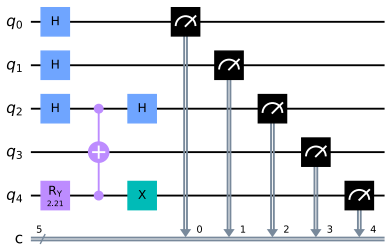

FIGURE A.13: The twenty-values number generator circuit. This image was taken from [264].

contiguous results. In this way, obtaining all the circuits necessary to simulate each dice used for RPGs is possible. Those circuits are depicted in Figs. A.11 ad A.13, while their execution outcomes are depicted in Figs. A.12 and A.14.

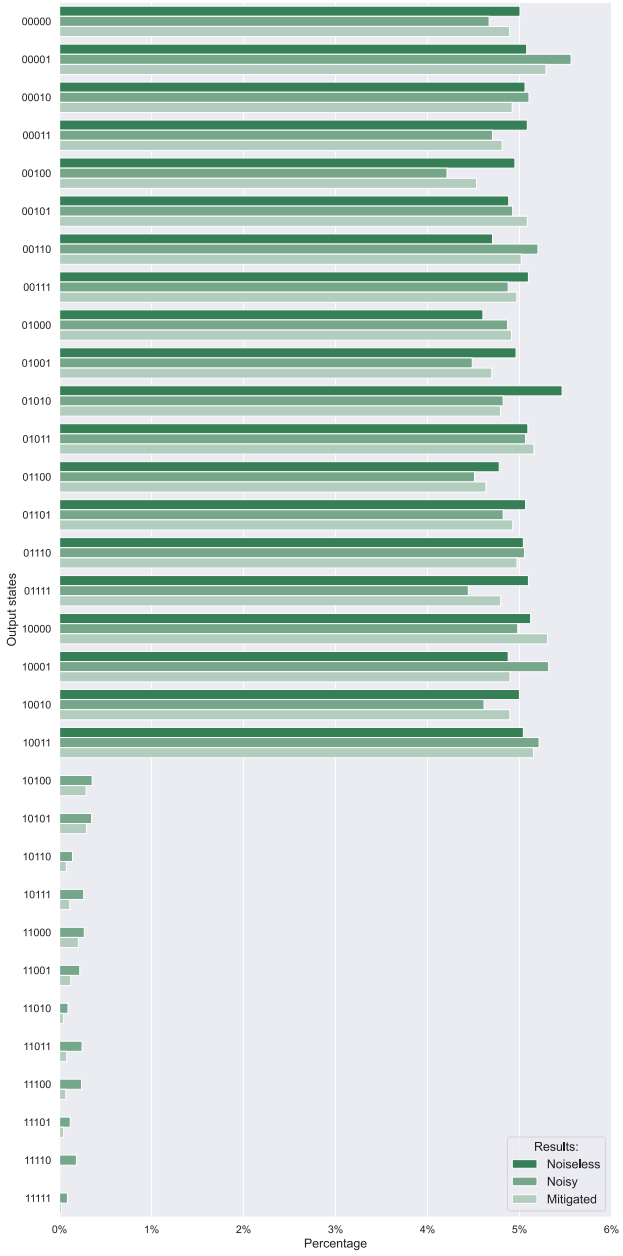

FIGURE A.14: The Noiseless, Noisy, and Mitigated outcomes acquired from executing the twenty-values number generator circuit for 16384 shots on the IBM Aer Simulator.

A.6.3 Interactive 3D Quantum Dice Roller

A dice rolling game, *Interactive 3D Quantum Dice Roller*, was created recurring to the quantum circuits described above [277]. This game is built on a modified version of Anton Natarov's 3D dice roller, where the number generation has been replaced with a quantum implementation [278] [279] . A screenshot of the game is shown in Figure A.15.

In the game, the user selects one or more dice to roll. The same die can be chosen several times by clicking on it repeatedly, while with the *Clear* button, removing a previously chosen die is possible. When the user clicks the *Throw* button, a loading interface hides the processing operations needed to run on actual or simulated quantum hardware. After receiving the results, an animation with the dice moving illustrates the random number generated by the quantum device. For convenience, the application is configured to run using the IBM QASM simulator, minimizing the wait time to a few seconds and avoiding errors arising from quantum noise. As demonstrated in the previous Sections, such noise is present in real devices and can create problems for circuits that use n qubits but require fewer than 2^n states, as unwanted states may result. To solve this problem, although imperfect, the

FIGURE A.15: A screenshot of the Interactive 3D Quantum Dice Roller game. This image was taken from [264].

application approximates the result to the highest value on the target die.
The name of the backend can be changed by entering the name of one of the IBM servers so as to run the game on real quantum hardware, such as IBMQ Jakarta, IBMQ Manila, IBMQ Quito, or others.

Acronyms

Symbols

10GigE 10 Gigabit Ethernet 19
3D Three Dimensional 3, 21, 27, 34, 35, 40–43, 53–56, 58, 135
4G Fourth Generation 29, 42, 135
5G Fifth Generation 135

A

AA Attention Array 108, 109, 111–113
AB Attention Behavior 111, 113, 114
AC Alternating Current 96
ACC Air Cargo Challenge 28
ADC Analog to Digital Converter 65
AG Attention Gain 111, 112
AI Artificial Intelligence 3, 4, 6, 90, 91, 105, 109
ANN Artificial Neural Network 73, 82, 87, 119–121, 123, 124, 132, 133
ANS Autonomic Nervous System 95, 97, 98, 101, 102
AP Average Precision 92, 94
AR Augmented Reality 27, 28, 39, 41, 48–52, 135
AU Action Unit 103

B

BB Beat-to-Beat 98
BCE Before the Common Era 5, 15
bit Binary Digit 1, 6, 8–10, 15
BLE Bluetooth Low Energy 108
BP Blood Pressure 97, 98, 101

BSON Binary JSON 33
BVP Blood Volume Pulse 100

C

CCD Charge-Coupled Devices 6, 60, 62, 66–68
CCNET Contractive Convolutional Network 104
CCNOT Controlled-Controlled-NOT 2
CFRP Carbon Fiber Reinforced Polymer 41, 48–50
CG C for Graphics 34
CK+ Extended Cohn-Kanade Dataset 105, 106
CMFD Correctly Masked Face Dataset 86
CMOS Complementary Metal-Oxide Semiconductor 6, 60, 62, 65, 67, 68
CNN Convolutional Neural Network 69, 73, 104, 106, 107
CNOT Controlled-NOT 2
CNS Central Nervous System 101, 102
COCO Common Objects in COntext 91
COVID-19 Coronavirus Disease 2019 27, 60, 69, 70, 79, 102, 110, 134, 136
CPU Central Processing Unit 41, 75, 76, 93, 94
CSV Comma-Separated Values 107
cuDNN CUDA Deep Neural Network 76, 94

D

DAUIN Department of Control and Computer Engineering 13, 25, 52
DBN Deep Belief Network 104
DC Direct Current 96
DDR Double Data Rate 75, 93, 94
DE Database Ensemble 104, 105
DET Department of Electronics and Telecommunications 26
DIATI Department of Environment, Land and Infrastructure Engineering 26
DIMEAS Department of Mechanical and Aerospace Engineering 13, 15, 26, 30, 40, 52, 134
DISAT Department of Applied Science and Technology 26
DL Deep Learning 76, 104
Dl4j Deeplearning4j 107
DSC Digital Still Camera 60

E

ECG Electrocardiogram 98, 100, 137
ED Emotion Detector 107
EEG Electroencephalography 101, 137
ELED Edge Emitting Light-Emitting Diode 18
ELM Extreme Learning Machine 104
EMI Electromagnetic Interference 19
EMPs Electromagnetic Pulses 19
EU European Union 117

F

FACS Facial Action Coding System 103
FBG Fiber Bragg Grating 14, 15, 23–26, 28, 29, 34–59, 134–136
FC Fully Connected 74, 122, 123
FEDC Facial Expressions Databases Classifier 105
FER2013 Facial Expression Recognition 2013 105
FF Fill Factor 61–63, 65, 67, 68
FFHQ Flickr-Faces-HQ3 86, 88, 89
FFT Fast Fourier Transformation 38
FMLD Face Mask Label Dataset 79, 82–85
FMR-DB Facial Mask and Respirator Database 69–73, 77, 78, 81, 136, 137
FNN Feed-Forward Neural Network 119, 122–124
FWC Full Well Capacity 62, 63

G

GA Grade of Attention 108
GDDR Graphics Double Data Rate 75
GHG Greenhouse Gas 117
GigE Gigabit Ethernet 19
GLSL OpenGL Shading Language 34
GM Game Master 17
GNU GNU's Not Unix 32, 33, 35, 39
GPL General Public License 32, 33, 35, 39
GPU Graphics Processing Unit 34, 35, 41, 75, 76, 93, 94
GUI Graphical User Interface 28, 30

H

HF High Frequency 98, 99
HLSL High-Level Shader Language 34, 35
HMI Human Machine Interface 117
HR Heart Rate 97, 98, 100–102, 108
HRV Heart Rate Variability 98–100, 108

I

IBI Interbeat Interval 99
ICNS Intrinsic Cardiac Nervous System 98
IEA International Energy Agency 117
IGBP International Geosphere-Biosphere Programme 2
IMFD Incorrectly Masked Face Dataset 86
IMU Inertial Measurement Unit 136
IoT Internet of Things 95, 100–102
IP Internet Protocol 30
IT Information Technology 27, 32, 57, 110, 136

J

JAFFE Japanese Female Facial Expression 105
JSON JavaScript Object Notation 33

K

KET Key Enabling Technology 26

L

LAN Local Area Network 19, 117
LARP Live-Action Role-Playing 16
LBP Local Binary Patterns 104
LBP-TOP Local Binary Patterns in Three Orthogonal Planes 100
LCL Lower Control Limit 121, 122, 125, 126, 128
LED Light-Emitting Diode 18, 22
LF Low Frequency 98, 99
LiPo Lithium-Ion Polymer 29
LR Learning Rate 74, 92, 123, 124

LSB Least Significant Bit 15
LSTM Long Short-Term Memory 122–124

M

MANOVA Multivariate Analysis of Variance 11
mAP Mean Average Precision 92, 94
Mask-RCNN Mask Region-based Convolutional Neural Network 91–93
ML Machine Learning 10, 13, 134
MMI Man-Machine Interface 117
MMORPG Massively Multiplayer Online Role-Playing Game 16
MOSFET Metal-Oxide Semiconductor Field-Effect Transistor 65
MSB Most Significant Bit 9, 10, 12
MSE Mean Squared Error 123, 124
MTCNN Multi-Task Cascaded Convolutional Neural Network 71
MTOW Maximum Takeoff Weight 28
MTU Master Terminal Unit 117
MUG Multimedia Understanding Group 105

N

NDA Non-Disclosure Agreement 92
NIR Near Infrared 19, 66
NN Normal to Normal 99
NN50 Number of Pairs of Adjacent NN Intervals Differing by More Than 50 ms 99
NoSQL Non-Structured Query Language 33

O

OPEX OPerational EXpenditure 118

P

PC Personal Computer 6, 34, 65, 66
Ph.D. Philosophiae Doctor 1, 23, 26–28, 134, 136
PLC Programmable Logic Controller 116, 117
pNN50 Proportion of NN50 Divided by Total Number of NNs 99
PP Peak-to-Peak 97, 99
PPG Photoplethysmography 14, 95–98, 137, 138

PRNG Pseudo-Random Number Generator 17, 18
PSN Public Switched Network 117

Q

QCLA Quantum Carry-Lookahead Adder 9
QCSA Quantum Carry-Save Adder 9
QR Quick Response 102
QRCA Quantum Ripple-Carry Adder 8, 9
qubit Quantum Binary Digit 1, 2, 4, 6–9, 11, 12, 14, 19, 21

R

RaFD Radboud Faces Database 105
RAM Random Access Memory 41, 75, 94
RAMS Reliability, Availability, Maintenance and Safety 120
RBI Residual Bulk Image 66
RELU Rectified Linear Unit 123
RES Renewable Energy Source 117, 118
RFI Radio-Frequency Interference 19
RFID Radio Frequency Identification 102
RMSE Root Mean Squared Error 111, 112, 123–125, 128
RMSSD Root Mean Square of Differences Between Adjacent Intervals 99, 108
RNG Random Number Generator 17
RNN Recurrent Neural Network 119, 123
ROI Return On Investment 118
RPG Role-Playing Game 1, 16–20
RPM Revolutions Per Minute 120
RPP Renewable Power Plant 118
RTU Remote Terminal Unit 116, 117

S

S Area of the Ellipse Representing Total Heart Rate Variability 99
SC Skin Conductance 100, 101
SCADA Supervisory Control And Data Acquisition 14, 116–118, 120, 122, 129, 131, 138
SD Standard Deviation 120, 121, 126

SD1 Standard Deviation Perpendicular the Poincaré Plot Identity Line 99
SD2 Standard Deviation Along the Poincaré Plot Identity Line 99
SDNN Standard Deviation of all Normal to Normal Intervals 99, 108
SDSD Standard Deviation of Differences Between Adjacent Intervals 99, 108
SFEW 2.0 Static Facial Expressions in the Wild 2.0 105
SGD Stochastic Gradient Descent 83
SIoT Social Internet of Things 102
SSD Solid State Drive 41, 75, 94
ST Short-Term 99

T

TCP Transmission Control Protocol 36, 38
TRPG Tabletop Role-Playing Game 16
TSO Transmission System Operator 121

U

UAV Unmanned Aerial Vehicle 15, 26, 28, 34, 35, 41, 51, 56, 57, 135, 136
UCL Upper Control Limit 121, 122, 125, 126, 128, 132
UDP User Datagram Protocol 29, 31
ULF Ultra-Low Frequency 99
USB Universal Serial Bus 29
UST Ultra-Short-Term 99
UV Ultraviolet 23

V

VCSEL Vertical Cavity Surface Emitting Laser 18
VIA VGG Image Annotator 91
VLF Very-Low Frequency 99
VOC Visual Object Classes 81

W

WAN Wide Area Network 117
WELL Well Equidistributed Long-period Linear 17
WS Window Size 108, 111–114
WTG Wind Turbine Generator 116, 118–133

WWMR-DB Ways to Wear a Mask or a Respirator Database 79–84, 86–88, 90, 137

X

XOR Exclusive OR 2, 17

Y

YOLO You Only Look Once 81