UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

DOTTORATO DI RICERCA IN INGEGNERIA MECCANICA E INDUSTRIALE

ING-INF/04 AUTOMATICA

XXXV CICLO

DESIGN AND DEVELOPMENT OF CONTROL STRATEGIES TO FACILITATE HUMAN-ROBOT INTERACTION

STEFANO GHIDINI

PROF. MANUEL BESCHI

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

HRI	Human Robot Interaction
pHRI	Physical Human Robot Interaction
VIC	Variable Impedance Control
EMG	Electromyogram
FIS	Fuzzy Inference System
SLS	Structured Light Scanner
EVM	Eulerian Video Magnification
APE	Absolute Position Error
AF	Absolute Force
ITAE	Integral Time Absolute Error
ITAEP	Integral Time Absolute Error Position
ITAF	Integral Time Absolute Error
SW	Software
GUI	Graphical User Interface
ROI	Region of Interest

ITALIAN ABSTRACT

L'industria 4.0 introduce una serie di paradigmi volti a portare cambiamenti radicali nel panorama industriale, risolvendo una serie di problematiche attualmente esistenti. Questi paradigmi sfruttano l'integrazione di sistemi digitali e fisici, con l'obiettivo di creare un sistema di produzione intelligente e completo basato su un rapido scambio di informazioni e raccolta dati. L'obiettivo di questa attività di ricerca è di indagare, analizzare e sviluppare una serie di strategie tecnologiche, inerenti ai pardigmi dell'industry 4.0, che vanno ad affrontare alcuni problemi legati al settore manifatturiero tramite due tecnologie differenti e che coprono altrettante aree di ricerca. In una prima parte del lavoro ci si è focalizzati su alcuni algoritmi di controllo per robot collaborativi nell'ottica di migliorare diversi aspetti dell'Interazione Uomo Robot, o Human-Robot Interaction (HRI). L'HRI è un aspetto centrale dell'Industria 4.0, e vede come uno dei suoi principali rami l'interazione fisica uomo robot (pHRI). Uno degli algoritmi di controllo più diffusi in questo campo è il controllo in impedenza. Esempi comuni di controllo in impedenza si trovano nelle applicazioni industriali: come la robot manual guidance o attività come il "peg-in-hole". In questa situazione, la dinamica dell'impedenza può essere configurata come un sistema massa-molla-smorzatore. L'elasticità della molla può essere vantaggiosa in un attività di "peg-in-hole", ma può essere meno adatta durante la robot manual guidance, dove una forza repulsiva può rendere più faticoso il lavoro dell'operatore. Tuttavia, nella pHRI, entrambi i compiti sono spesso svolti contemporaneamente. Per questa ragione sono stati realizzati due algoritmi di controllo basati su una variazione dinamica dell'impedenza.

Il primo algoritmo presenta una modalità operativa plastica dedicata, che consente la deformazione plastica della traiettoria del robot quando viene applicata una forza esterna. Grazie a questo controllo elasto-plastico, superati certi limiti il comportamento di pura impedenza viene quindi sostituito da un comportamento plastico, consentendo il movimento del robot verso altre posizioni. Applicazioni come la manual guidance si prestano meglio al controllo plastico, poiché il robot deve essere spostato in posizioni fisse in modo fluido. Altre applicazioni, come il "peg-in-hole", preferiscono il controllo classico dell'impedenza, poiché il robot deve deviare elasticamente dalla sua traiettoria durante il compito in modo da consentire un preciso ma flessibile posizionamente del pezzo manipolato.

Il controllo dell'impedenza può anche essere totalmente variabile, definendo il cosiddetto controllo a impedenza variabile, o Variable Impedance Control (VIC). I ricercatori hanno studiato l'adattamento dell'impedenza

basandosi su precedenti analisi dell'impedenza del braccio umano durante attività di pHRI. Tuttavia, queste soluzioni sono fortemente legate a un modello preciso del braccio umano, che non sempre è facilmente ricavabile. Considerando ciò, l'algoritmo di controllo sviluppato si basa su un modello della catena uomo-robot che non richiede una precedente identificazione dell'impedenza del braccio umano ma si basa su un modello matematico totalmente offline per descrivere l'operatore umano e la sua attività. Sulla base di questo modello verrà realizzato un controllo Fuzzy che modulerà i valori di impedenza al fine di migliorare i risultati di un'attività di pHRI, intesi come buona accuratezza posizionale del robot e basso sforzo fisico richiesto dall'operatore.

Nella seconda parte delle tesi sono state realizzate applicazioni per il monitoriaggio della qualità in ambito industriale basate su tecniche di Computer Vision. L'ispezione, la misurazione e il rilevamento dei difetti sono procedure di controllo cruciali, ampiamente utilizzate nei processi industriali. Una prima applicazione usa uno scanner 3D per acquisire un pezzo da analizzare. Sul mercato esistono diversi tipi di scanner laser 3D, tuttavia si tratta di stazioni di misura e metrologia complesse in cui un componente può essere analizzato a campione o alla fine del processo di produzione. D'altro canto, un controllo diretto in tempo reale sulla linea di produzione permetterebbe il monitoraggio a monte di eventuali difetti presenti sul componente. In quest'ottica è stato sviluppato un prototipo di software per implementare un sistema di ispezione in linea di produzione che cerca di compensare questa mancanza.

In seguito è stata sviluppata un'altra applicazione basata su Computer Vision usando l'algoritmo *EVM*, un algoritmo utilizzato per rivelare piccole variazioni di segnale in una registrazione video, e che non sono facilmente visibili ad occhio nudo. Questo ha permesso di identificare, tramite una videocamera: battito cardiaco e respirazione di una persona (in ambito healthcare) e vibrazioni meccaniche prodotte da macchinari (in ambito industriale e manifatturiero).

ENGLISH ABSTRACT

Industry 4.0 introduces a series of paradigms aimed at bringing about a change in the industrial field by solving a number of currently existing problems. These paradigms focus on the integration of digital and physical systems, with the goal of creating an intelligent and comprehensive manufacturing system based on rapid information exchange and data collection. The objective of this research activity is to investigate, analyze, and develop a series of technological strategies inherent to the paradigms of Industry 4.0 that address problems related to the manufacturing sector through two different technologies and covering as many research areas.

In the first part of the paper, we focused on control algorithms for collaborative robots with a view to improving aspects of Human-Robot Interaction (HRI), which is a central aspect of Industry 4.0 and sees physical Human-Robot Interaction (pHRI) as one of its branches. One of the most popular control algorithms in this field is impedance control, which can be configured as a mass-spring-damping system. Spring elasticity may be advantageous in a peg-in-hole task, but may be less suitable during, for instance, robot manual guidance, where a repulsive force may make the operator's work more strenuous. However, in pHRI, both tasks are often performed simultaneously. For this reason, two control algorithms based on dynamic impedance variation have been realized.

The first algorithm has a dedicated plastic operating mode, which allows plastic deformation of the robot's trajectory when an external force is applied. In the proposed elasto-plastic control, beyond certain limits, pure impedance behavior is replaced by plastic behavior, allowing permanent movement of the robot to a new position. Applications such as manual guidance lend themselves better to plastic control, since the robot must be moved to fixed positions smoothly. Other applications, such as "peg-in-hole," prefer classical impedance control, since the robot must elastically deviate from its trajectory during the task to allow precise but flexible positioning of the manipulated workpiece.

Impedance control can also be totally variable, defining what is known as Variable Impedance Control (VIC). The researchers have studied impedance adaptation based on previous analyses of human arm impedance, in the context of pHRI activities. However, these solutions are strongly related to an accurate model of the human arm, which is not always easily derived. Considering this, the developed control algorithm is based on a human-robot chain model that does not require previous identification of the human arm impedance but relies on a totally offline mathematical model to describe the human operator and its activity. Based on this model, a

Fuzzy control is designed in order to modulate impedance values with the aim of improving the results of a pHRI task, defined as good positional accuracy of the robot and low physical effort required by the operator.

In the second part of the thesis, applications for quality monitoring in industrial settings based on Computer Vision techniques were implemented. Inspection, measurement, and defect detection are crucial control procedures that are widely used in industrial processes. A first application uses a 3D scanner to acquire a part for analysis. There are several types of 3D laser scanners on the market, however, these are complex measuring and metrology stations in which a part can be analyzed either by sample or at the end of the manufacturing process. On the other hand, direct real-time monitoring on the production line would allow upstream monitoring of any defects present on the component. With this in mind, a prototype software to implement a production line inspection system that attempts to compensate for this deficiency was developed.

In conclusion, another Computer Vision-based application was investigated and developed. This application uses the *EVM*, an algorithm used to reveal small variations in a video recording that are not easily noticeable to the naked eye. This made it possible to identify, via a camera: a person's heartbeat and respiration (in healthcare) and mechanical vibrations produced by machines (in industrial and manufacturing).

LIST OF PUBLICATIONS

The main research topic of this thesis was the development of robot's control algorithms related to the Human-Robot Interaction. In addition, we have covered other researches related to the development of hardware/software applications to improve some aspects of the the quality control in the manufacturing field using optical sensors and computer vision techniques.

The theoretical and practical results achieved in both topics have generated the following publications:

1. **Ghidini, S.**, Beschi, M., & Pedrocchi, N. (2020). A robust linear control strategy to enhance damping of a series elastic actuator on a collaborative robot. Journal of Intelligent Robotic Systems, 98(3), 627-641.

2. Nuzzi, C., **Ghidini, S.**, Pagani, R., & Ragni, F. (2020, March). RemindLy: A Personal Note-bot Assistant. In Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction (pp. 631-632).

3. Nuzzi, C., **Ghidini, S.**, Pagani, R., Pasinetti, S., Coffetti, G., & Sansoni, G. (2020, June). Hands-Free: a robot augmented reality teleoperation system. In 2020 17th International Conference on Ubiquitous Robots (UR) (pp. 617-624). IEEE.

4. Faroni, M., Beschi, M., **Ghidini, S.**, Pedrocchi, N., Umbrico, A., Orlandini, A., & Cesta, A. (2020, August). A layered control approach to human-aware task and motion planning for human-robot collaboration. In 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) (pp. 1204-1210). IEEE.

5. Nuzzi, C., Pasinetti, S., Pagani, R., **Ghidini, S.**, Beschi, M., Coffetti, G., & Sansoni, G. (2021). MEGURU: a gesture-based robot program builder for Meta-Collaborative workstations. Robotics and Computer-Integrated Manufacturing, 68, 102085.

1 INTRODUCTION

1.1 INDUSTRY 4.0 CONCEPTS

Industry 4.0 refers to a revolutionary shift in the industrial field, focused on the integration of cyber and physical systems. The seamless connection of digital and physical systems, along with advanced technologies, is poised to transform the way work is performed and managed. This shift has the potential to disrupt, or at least alter, the traditional balance of cost, flexibility, speed, and quality in industrial operations. Industry 4.0 promotes the idea of a number of technologies that enable an almost real-time connection between the physical and digital realms, aiming to establish a comprehensive, intelligent smart manufacturing system for interconnected industrial information applications [5]. Nine pillars define the main drivers that lay the foundation for industry 4.0:

- *Big Data and Analytics*: The collection and comprehensive evaluation of data from various sources, such as production equipment and systems as well as enterprise and customer management systems [6];

- *Autonomous Robots*: Robots are becoming increasingly autonomous, flexible, and cooperative, will eventually interact with one another and work safely alongside humans, learning from them in Human-Robot interaction tasks [6];

- *Simulation*: Simulations will be used more extensively in plant operations to leverage real-time data to mirror the physical world in a virtual model 2D and 3D models. This can include machines, products, and humans [6];

- *System Integration*: Integration and self-optimization are the two major mechanisms used in industrial organization [7]. This includes horizontal integration, across the entire value creation network, and vertical integration;

- *The Industrial Internet of Thing (IIoT)*s: The Internet of Things refers to a worldwide network of interconnected and uniformly addressed objects that communicate via standard protocols [8];

- *Cybersecurity and Cyber Physical Systems (CPS)*: With the increased connectivity and use of standard communications protocols that come with Industry 4.0, the need to protect critical industrial systems and manufacturing lines from cybersecurity threats increases dramatically [6]. Secure and reliable communications, as well as sophisticated identity and access management of machines and users, are essential;

- *The Cloud*: Cloud-based IT platforms serve as the technical backbone for connecting and communicating various elements of the Application Centre Industry 4.0 [9];
- *Additive Manufacturing*: with Industry 4.0, additive manufacturing methods will be widely used to produce small batches of customized products that offer construction advantages, such as complex, lightweight designs. High-performance, decentralized additive manufacturing systems will reduce transport distances and stock on hand [7];
- *Augmented Reality*: Augmented reality-based systems support a variety of services, such as selecting parts in a warehouse and sending repair instructions over mobile devices. Industry can use augmented reality to provide workers with real-time information to improve decision making and work procedures.

Nine fundamental elements of Industry 4.0 will revolutionize the way isolated and optimized production units function by converting them into a completely integrated, automated, and optimized production process. This results in increased productivity and alters the traditional production connections between suppliers, manufacturers, and consumers as well as between human and machine [6]. Moreover, the four main drivers of Industry 4.0 are Internet of Things (IoT), Industrial Internet of Things (IIoT), Cloud based manufacturing and Smart Manufacturing which helps in transforming the manufacturing process into fully digitized and intelligent one [10]. With this in mind, the goal proposed in this research activity is to investigate, analyze, and develop a range of innovative strategies which go to address some problems related to manufacturing field by means of two different technologies that cover as many different research areas.

The first area investigates different robot's control algorithms based on impedance control to facilitates human-robot interaction (HRI) and, in general, human-machine interaction (HMI). There are several industrial task, such as assembly and disassembly applications, where the operator has a key role within the manufacturing cell and must cooperate closely with the robot.

The second research area concerns the development of different applications (hardware and software) based on optical sensors and computer vision, in order to improve the quality control in manufacturing processes. In fact, some key features of Industry 4.0 include real-time data monitoring, tracking products, and mostly, controlling production processes [11].

1.2 IMPEDANCE CONTROL

The field of Human-Robot Interaction (HRI) is an important area of study in both academia and industry [12]. One of the key challenges within HRI is in the realm of physical interaction between humans and robots, known as pHRI. In this area, robots interact with humans in shared workspaces [13, 14]. Applications of pHRI can be found in industries such as manufacturing [15, 16, 17], rescue operations [18], and rehabilitation [19]. In the case of a pHRI, a proper and safe set of control algorithms for the robot is necessary, since the robot physically comes into contact with the human and may cause collisions or damage.

The primary methods for controlling robots are known as motion control and force control [20]. The goal of motion control is to guide the robot along a specific path in space, while force control focuses on commanding the force applied by the robot on the environment. Unfortunately, neither of these methods is suitable when a human operator is present within the robot's workspace and, therefore, to pHRI applications. In motion control, all obstacles encountered, whether human or object, are considered disturbances, human or object, are considered disturbances and require increased force to avoid the obstacle. In contrast, force control only uses positioning to achieve the necessary conditions to apply the intended force.

One popular approach to overcome this difficulties is called [21, 22, 23] This method involves the regulation of the robot's impedance and dynamic behavior - including mass, damping, and stiffness - in order to achieve a more natural and safe physical interaction, minimizing the risk of damage in the event of unexpected collisions with the humans or the surroundings [24]. Due to its ability to adapt to the environment, impedance control is frequently employed in collaborative applications [25, 26] where humans and robots work together on a shared goal, and ordinarily in situations where physical interactions between humans and robots are necessary [27, 28]. In this case, mechanical impedance is the relation between velocity of the robot and the force exchanged. In Laplace domain, it can be expressed as [29]:

$$Z(s) = \frac{F(s)}{\dot{X}(s)} = \frac{F(s)}{sX(s)} \tag{1}$$

It is simple to observe how Z(s) allows for the decoupling of force and position terms. A frequent method to apply impedance control is to consider a spring-mass-damper model, indeed, by rearranging the terms:

$$F(s) = sZ(s)X(s) \tag{2}$$

$$F(s) = ms^2 + cs + k \tag{3}$$

Obtained by the following impedance:

$$Z(s) = ms + c + \frac{k}{s} \tag{4}$$

where m,c and k are, respectively, mass, damping and elastic constants.

Common instances of impedance control can be found in industrial application ([30, 31]), such as robot manual guidance or assembling parts using a peg-in-hole [32]. In this situation, the impedance dynamics can be configured as a mass-damper system. It is important to note that the stiff behavior provided by the spring can be beneficial in the latter example since the elasticity introduced by the spring helps in positioning task, but may not be suitable for robot manual guidance where a repulsive force can make the operator's work more strenuous. However, in pHRI, both tasks are often performed simultaneously and under tight time constraints. Therefore, an impedance control algorithm that can adapt itself on the basis of predetermined conditions or which is capable of smoothly change from elastic to plastic deformation could improve efficiency and user satisfaction. For this reason, the present work proposes two types of control system based on impedance control: an *Elasto-plastic control algorithm* and a *Fuzzy-based Variable impedance control Algorithm*.

1.2.1 *Elasto-plastic control algorithm*

This control algorithm presents a plastic operating mode, which enables plastic deformation of the robot's trajectory when an external force is applied. A plastic behaviour occurs when, as a result of applying a force on the system, there is a permanent change in position. A typical impedance control is beneficial for minor deviations from the reference due to forces of brief duration. However, when significant position deviations are needed, the force generated by the controller as a response to external forces can become significant; in fact, due to the elastic term $k\Delta x$, the greater the position variation, the greater the force opposed. In the proposed elasto-plastic control, beyond certain limits the pure impedance behavior is replaced by a plastic behavior, allowing permanent movement of the robot to a new position. Applications such as manual guidance are best performed with plastic control, as the robot needs to be moved to fixed positions in a smooth manner. Other applications, such as peg-in-hole, prefer classical impedance control, as the robot must deviate elastically from its reference during the task. When both activities are performed during the same session and under strict time constraints, this a "two-mode control algorithm" is helpful in increasing human operator satisfaction.

1.2.2 *Fuzzy-based variable impedance control Algorithm*

Impedance control is also used by regulation of the robot mechanical impedance in response to a desired reaction during a specific task. Indeed, the so-called, called Variable Impedance Control (VIC) involves adjusting the values of components such as stiffness, damping, and mass during task execution using a specific control strategy [33, 34, 35]. Researchers have studied adapting impedance based on previous analyses of human arm impedance [36, 37] and others have considered impedance control while incorporating a general model of the HRI [38, 39]. More recent developments include utilizing variable admittance control strategies and new fuzzy logic learning methods to manage the robot's impedance characteristics [40].

Nevertheless, these solutions are strongly related to a precise model of the human arm, a specific knowledge of the task, or initial robot learning phase. Moreover, many human-arm models are derived from electromyogram analysis (EMG) [41, 42]. However, EMG could be affected by difficulties in processing EMG signals for amplitude and spectral analysis, reducing their efficiency for many applications. From this perspective, it can be evinced how a more simpler HRI model could reduce the level of the problems occurred, if the task fit proper requirements such as, for instance, simplicity of execution. Consider the problems listed above, the control algorithm proposed is based on a model of the human-robot chain which does not require a previous identification of the human arm impedance or a detailed knowledge of the task, instead, it is based on a simpler quasi-linear mathematical model to describe the human operator and its activity with a totally offline method, as the one investigated by McRuer [43]. To achieve this, a fuzzy-based control strategy has been developed. This control is a fuzzy-based with a variable Cartesian impedance control which is able to adapt online parameters such as mass, stiffness and damping by exploiting a simple HRI model as proposed in [43]. The main goal of the developed structure is to speedup industrial tasks that require pHRI, while keeping accuracy and user's satisfaction.

On this basis, taking into account a manual guidance operation where the user physically moves the robot, the impedance parameters are scaled-up in order to cover rapid movements (robot approaches close or far from the target), and slow movements for precise positioning. A Fuzzy Inference System (FIS) is used in combination with an optimization algorithm to identify an input-output fuzzy surface that suitably scales impedance parameters by reducing a cost which includes the human absolute force (AF) applied on the end-effector, and the absolute Cartesian position error (APE).

1.3 VISION SYSTEMS FOR QUALITY CONTROL

In line with Industry 4.0 and Smart Manufacturing principles, modern industrial systems should implement a range of processes that integrate traditional industrial automation with digital technologies. Additionally, there is a need for cost-effective digital automation techniques that can be seamlessly integrated into existing production processes.

Inspection, measurement, and fault detection are crucial control procedures widely used in industrial assembly processes [44]. Currently, the inspection of assembled units is not automated and is typically carried out manually by operators using specialized tools such as steel rules, micrometers, and calibers, which require direct physical contact. This approach has several drawbacks, including being time-consuming, only able to be applied to one sample out of the production process, and subject to human error. Moreover, smart factories of the fourth industrial revolution are conceived by combining the tangible realm of manufacturing equipment with the digital realm of information and communication technology. In this scenario, manufacturers must be cognizant that producing a single item must remain economically viable. Consequently, transformative changes in intelligent machinery and other sophisticated equipment on the factory floor must be communicated by smart quality control to guarantee the delivery of superior goods to clients. Customers also wish to have access to immediate quality data to confirm that the final products meet their specifications. To accomplish this task, vision systems are a viable alternative. They act without contact, so there is certainty of not altering the product to be analyzed or the measurement. They can accelerate quality inspection procedures with high precision and traceability if necessary. Such system can examine the geometric features and extract some useful metrology information, significantly enhancing productivity within the manufacturing process.

1.3.1 *3D scanning techniques*

Currently, the use of non-contact measurement systems is increasing in metrology because of their effectiveness when measuring parts that cannot be measured using coordinate measuring machines or can only be measured with difficulty. Among non-contact technologies, 3D scanning techniques can best fit this task. Unlike coordinate measuring machines, 3D scanning techniques are not as suitable for performing such a wide range of measurements with high precision. However, in certain situations it may be more advantageous to utilize non-contact optical measuring systems rather than utilizing methods that require physical contact. There are scanners of varying designs possess appropriate uses based on their dimensions, costs, operating ranges, spatial precision, exactness, and scanning zones. For instance, utilizing a method of active triangulation scanning to examine the shape of worn cutting inserts, as seen in [45], assessing the wear of milling tools through an optical method, as shown in [46],

Since the 1980s, 3D scanners have evolved into many forms such as photogrammetry [47] time-of-flight [48], laser triangulation [49],and interferometry [50].

1.3.1.1 *Photogrammetry*

Photogrammetry exploits computer vision and mathematical algorithms to create 3D models from 2D photographs. These models are constructed by matching distinct features seen in pictures taken from various positions and angles. This approach is cost-efficient as it only requires a single camera. However, it has limitations in terms of precision (measured in millimeters) and the time required for reconstruction (measured in hours), making it primarily suitable for use in the mapping of buildings, landscapes, or for hobby photography. It is not practical for use in the 3D printing process as it requires a large number of photographs taken from various angles to produce a high-quality scan.

1.3.1.2 *Time-of-flight (TOF)*

This technology is based on a camera that determines the distance between the surface being measured and the device by measuring the time it takes for a laser to travel there and back. However, this method is only suitable for measuring larger objects as the precision is not very high (measured in millimeters). Additionally, the duration of the measurement is directly related to the size of the area being measured, as only one point of data can be obtained at a time.

1.3.1.3 *Interferometry*

Interferometry is a method of surface measurement with extremely high precision (measured in nanometers) that utilizes the interference patterns of two parallel light beams. The pattern produced is dependent on the distance between the light source and the object being examined. While the accuracy of this method is unparalleled, the equipment required is quite large and the scanning process is quite slow, covering only a small area with each scan.

1.3.1.4 *Laser triangulation*

Triangulation involves utilizing three reference points to calculate the shape of a surface. These points include the source of light, a spot on the surface being measured, and the device that detects the light. The distance between the surface and the scanner affects where the light spot appears on the sensor. When the surface has variations in height, the spot's position on the sensor will also change in relation to these variations. Triangulation can be performed using either a line or dot scanning method. Line scanning generates multiple data points, while dot scanning only generates one. The precision of these methods is restricted by factors such as the width of the line and the size of the laser spot, which is typically on the sub-millimeter level. Additionally, the equipment required for triangulation can be costly as it includes a high-end laser. Although line scanning is more efficient than dot scanning, it is still not fast enough for use in manufacturing application as it only produces one line of data at a time.

1.3.1.5 *Structured light scanner (SLS)*

Structured light scanner (SLS) is another approach to triangulation. Instead of using a laser to emit dots or lines, like in laser triangulation, SLS employs a projector that projects fringe patterns onto the surface being measured (Figure 1 and Figure 2). With just one scan, which takes several seconds, it can capture the entire projected area, making it a faster option for data collection compared to laser triangulation. The size of the area covered can be altered by adjusting the focus of the camera and projector to the desired field of view (FOV). Due to limitations in hardware size and shape, the FOV is restricted to centimeters, resulting in a sub-millimeter level spatial resolution . A smaller FOV will increase the spatial resolution, but it also poses new challenges in terms of system design and calibration.

SLS is the optimal choice for monitoring manufacturing process due to its affordability, compact design, and efficient scanning speed. On the contrary, photogrammetry and TOF, are more inaccurate and lacks the ability to collect sufficient data. Interferometry boasts high precision, but its slow scanning speed and large size make it unsuitable for integration within metal AM machinery. Laser triangulation, while precise, is limited to line scanning, rendering it inadequate for rapid area scanning. There are various types of 3D laser scanners on the market that are used in industry, such as the GOM ATOS Scan [52]. However, these constitute complex measuring and metrology stations, where a component can be analyzed

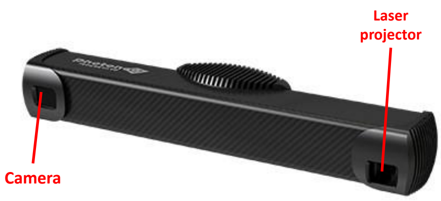


Figure 1: 3D laser scanned used. With a size of 77x68x416[mm] it can scan an area of 590x404[mm] at the sweat spot with an accuracy up to 0.1[mm].

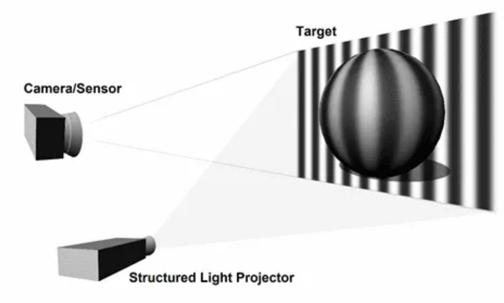


Figure 2: Projection of a laser pattern on the surface of the object. the deformation induced by the surface of the object is acquired by a camera and exploited for the calculation of the three-dimensional coordinates [51].

by sample or at the end of the production process. On the other hand, a direct real-time control on the production line would allow upstream monitoring of any defects present on the component. In this thesis, a prototype software was developed to implement an in-line production inspection system that solves this purpose.

1.3.2 *Eulerian Video Magnification (EVM)*

The Eulerian video magnification (EVM) is an algorithm employed to reveal subtle changes or hidden signals in videos that are not easily noticeable to the naked eye and present them. The movements that are too subtle for human perception, with amplitudes below our visual threshold, can provide valuable insights. Two fields benefit the most from this technology: manufacturing and healthcare. For instance, in the healthcare field, such invisible variations can be the fluctuations of respiratory motion or the changes in skin color due to blood flow as shown in the example of Figure 3, and can be used to extract important information, such as pulse rate [53, 54].

In manufacturing it is possible to place a camera in front of an engine or a moving component; through a video recording it is possible to amplify the vibrations in the frequencies of interest, resulting in a video signal in which the amplified vibration of interest can be observed. These amplitude movements can be analyzed to reveal mechanical behavior of the component [55].

High-speed videos of small eye movements, engine vibrations, and long-term physical processes like the melting of glaciers or growth of plants can also reveal important biological and physical changes. These

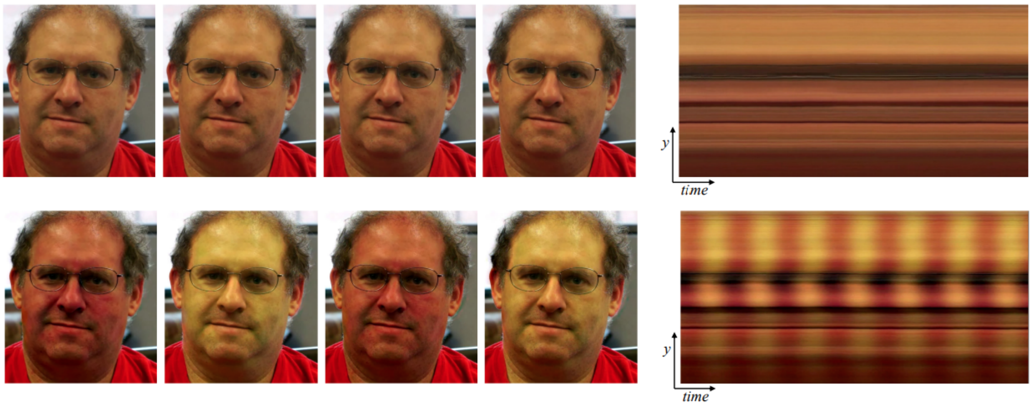


Figure 3: Eulerian Video Magnification method to detect the human heartbeat. Top: Four frames from the initial video recording, and a vertical scan line of the input plotted over time. Bottom: the same four screenshots with the subject's heartbeat signal intensified, and a vertical intesified scan line of the input plotted over time. In the input sequence the signal is unnoticeable, but in the magnified sequence the fluctuation is obvious. [4]

low amplitude movements can be amplified to reveal mechanical behavior. Video motion magnification is useful in various fields, including medical and scientific research, non-invasive vital sign monitoring, photography, and monitoring the structural integrity of buildings, bridges, and railroads during search and rescue operations.

As illustrated in [4], the process of this method employs a combination of spatial and temporal techniques to highlight delicate temporal variations in a video. The procedure is demonstrated in Figure 4.

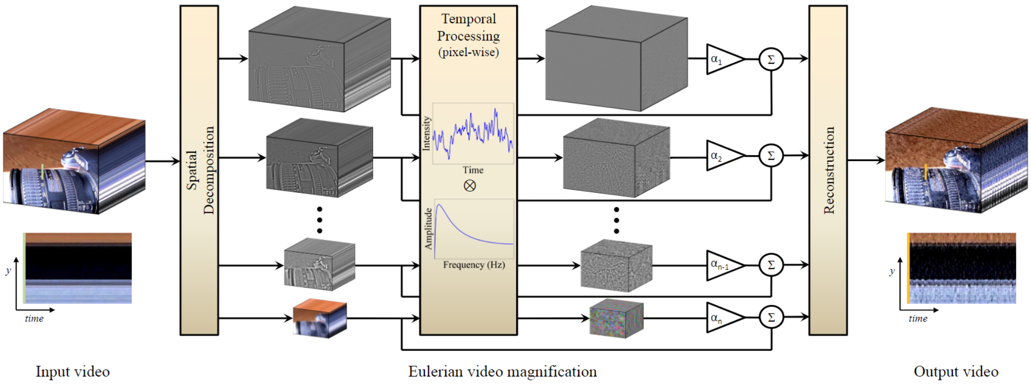


Figure 4: EVM overviw. The algorithm decompose the input video sequence into differentspatial frequency bands, and applies the same temporal filter to all bands. The filtered spatial bands are then intensified by a specific factor α, added to the original signal, and merged to create the final output video. The temporal filter and amplification factor can be adjusted to fit different purposes. [4]

Initially, the video sequence is decomposed into different spatial frequency bands. These bands may be amplified differently due to either different signal-to-noise ratios or because they contain spatial frequencies and we decrease the amplification for these bands to eliminate artifacts. When the goal of spatial processing is to simply boost temporal signal-to-noise ratio by combining multiple pixels, we spatially low-pass filter the frames of the video and downsample them for computational efficiency. In the general case, a full Laplacian or Gaussian pyramid is computed. Then, a temporal processing on each spatial band is carried out. We take into account the time series corresponding to the value of a pixel in the desired frequency band and apply a bandpass filter to extract the frequency bands of interest. For instance, we might choose frequencies within $0.4 - 4[\mathrm{Hz}]$, corresponding to $24 - 240[\mathrm{bpm}]$, if we want to amplify the heartbeat. We then multiply the extracted bandpassed signal by a magnification factor α. This factor can be specified by the user. At this point, the amplified signal is summed to the original frame and the spatial pyramid is recomposed to obtain the final output.

in this thesis, an industrial software (SW) application implementing this technique has been developed. The software has been designed to be plug-and-play and easily usable by an operator. Furthermore, in a context influenced by COVID-19, it was decided to give particular importance to the healthcare part, developing a system for monitoring a person's heartbeat and breathing, in order to have a small SW platform where the user is able to carry out a self-diagnosis and keep track of its health.

2 IMPEDANCE CONTROL ALGORITHMS

In the natural world, materials often exhibit elasto-plastic characteristics [56]. Among them, friction was selected as the base of the controller. Friction is a well-established subject, commonly utilized in control theory, with various models having been developed [57]. Models based on the bristle analogy, derived from Dahl's [58] model, emphasize the desired elasto-plastic behavior. One of this, the LuGre friction model [59, 60] is a one-dimensional friction model that effectively captures the majority of friction properties with a minimal number of adjustable parameters, reproducing an elasto-plastic behavior: spring-like behavior for small displacements and plastic behavior for greater ones.

In this first work, we created a control system based on the dynamic behavior of friction that results in elasto-plastic motion for the robot. We extended the traditional 1D friction model to a 3D model, thus enabling elasto-plastic control in the robot's 3D Cartesian space. To overcome the issue of an elastic component remaining present even during the plastic phase of the control, we proposed two distinct reset strategies that allow for purely plastic behavior to precisely achieved in every scenario.

2.1 ELASTO-PLASTIC FRICTION MODEL

Taking into account the LuGre friction model, it visualizes the contact between two surfaces as many bodies that make contact through elastic bristles. When a tangential force is applied, the bristles will deflect like springs which gives rise to the friction force. If such force is adequately large, some bristles could deflect so much that they will slip, leading to a pure sliding regime. The 1D LuGre friction model has been well described in [59]. Denoting the relative position between the two surfaces as x and the relative velocity as $\dot{x}$, the LuGre model can be derived by the following equations:

$$\begin{cases} F_d = \sigma_0 z + \sigma_1 \dot{z} + c_1 \dot{x} \\ \dot{z} = \dot{x} - |\dot{x}| \frac{\sigma_0}{g(\dot{x})} z \\ g(\dot{x}) = c_0 + c_s e^{-h|\dot{x}|} \end{cases} \tag{5}$$

Where, $F_d(z, \dot{x})$ is the friction force generated from the bending of the bristles, and comes from a z-dependent spring-damping model, with parameters σ_0 and σ_1, plus a damping term on the relative velocity which represents the viscous friction, with coefficient c_1. The second equation describes the evolution of z, a state variable that represents the bristles displacement during the motion. At the steady state, z approaches the value:

$$z_{ss} = \frac{g(\dot{x})}{\sigma_0} \frac{\dot{x}}{|\dot{x}|} = \frac{g(\dot{x})}{\sigma_0} sgn(\dot{x}) \tag{6}$$

Additionally, $g(\dot{x})$ is a function which expresses the sum of a Coulomb friction value c_0 and the Stribeck term $c_s e^{-h\dot{x}}$. Since the aim is only to replicate and elasto-plastic behaviour, the third equation in 5 related to $\dot{z}$ has been linearized by neglecting the term c_s, obtaining $g(\dot{x}) = c_0$; indeed, the Stribeck term is not required for an elasto-plastic behaviour. A simple graphic of the LuGre model compared with the Dahl model is illustrated in Figure 5.

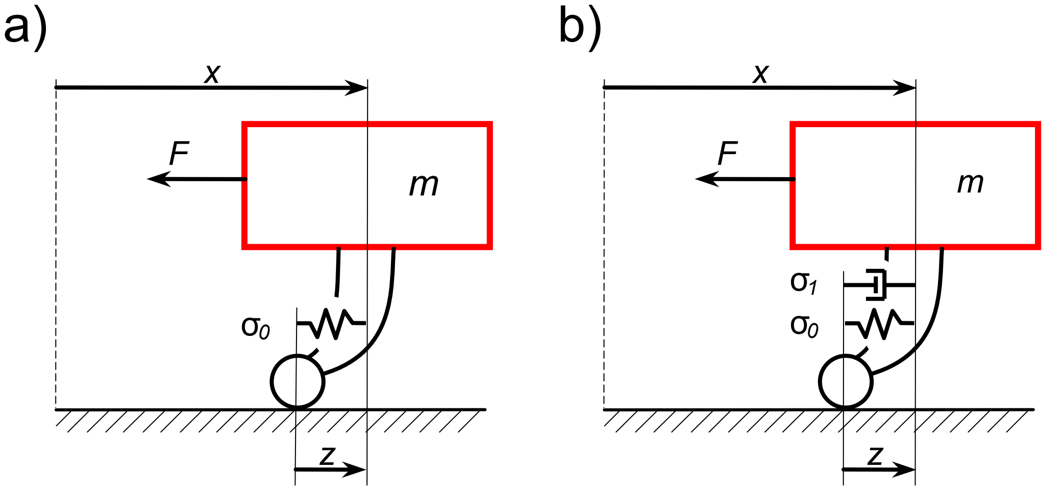


Figure 5: variable state z of: a) Dahl model (left) compared with LuGre model (a). Mass and bristles are represented here.

In Figures 6 and 7 the response of a unit of mass when the LuGre model is used to simulate friction. In the plot of Figure 6, a steady force of 0.2 [N] is applied for a period of 5 [s] before being removed. This plot emphasizes the elastic properties of the model. The mass reaches a stable position and, once the force is removed, the elastic component returns the mass to its initial position. In the plot of Figure 7, a similar force pattern is applied, but with a force of 40N. In this scenario, the state variable reaches its maximum limit. The position increases gradually until the external force is removed. In this case, the elastic component is not strong enough to return the mass to its initial position.

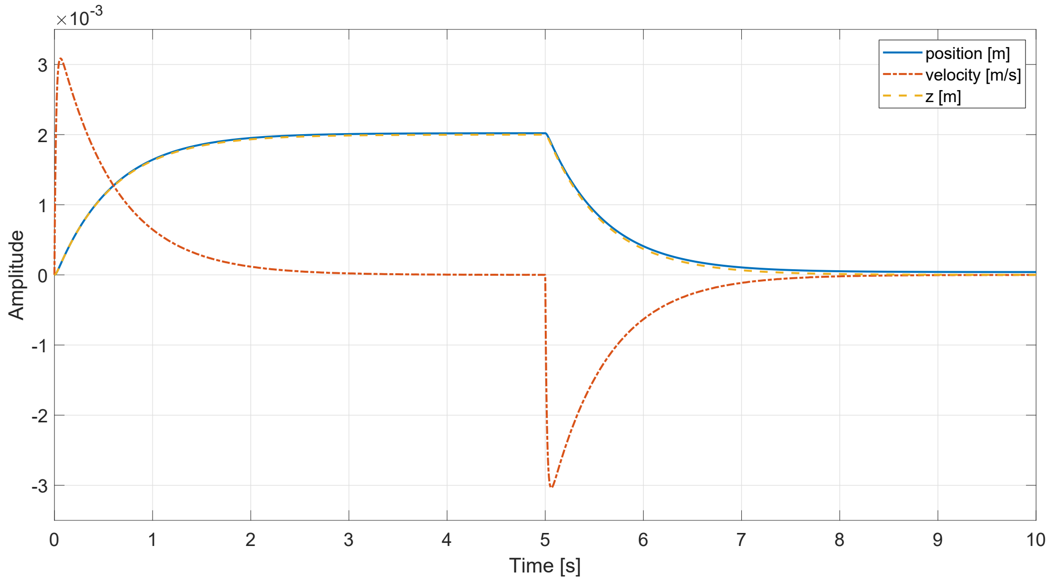


Figure 6: (a) LuGre elastic behavior. External force $F = 0.2$ [N] applied for 5 [s]. Parameters: $\sigma_0 = 100\frac{[N]}{m}$, $\sigma_1 = 50[\frac{Ns}{m}]$, $c_0 = 10[N]$, $c_1 = 10[\frac{Ns}{m}]$.

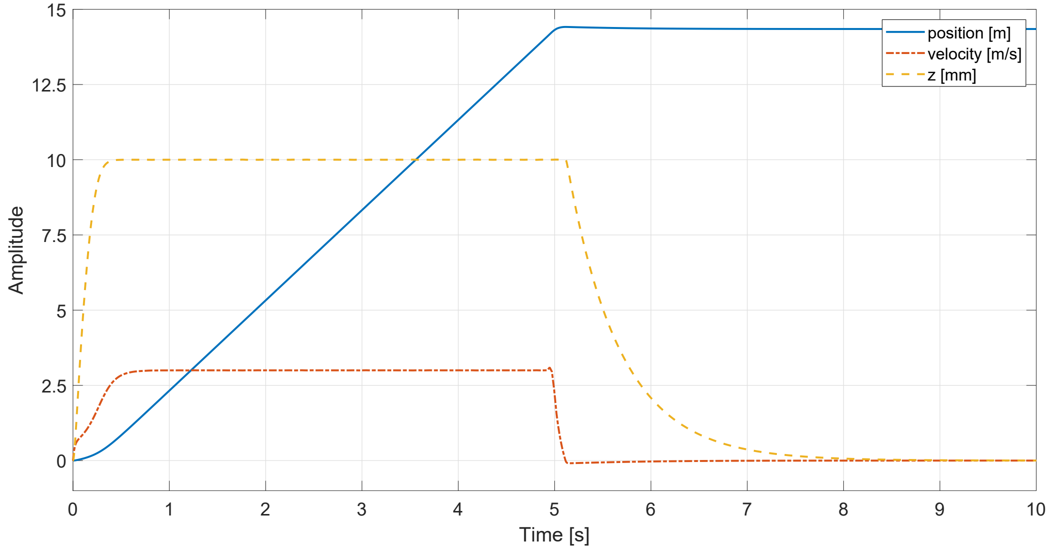


Figure 7: LuGre plastic behavior. External force F = 40 [N] applied for 5 [s]. The system's position is represented by the blue line, its velocity is indicated by the dotted red line, and the deformation of the bristles is represented by the dashed yellow line. Parameters: $\sigma_0 = 100\frac{[N]}{m}$, $\sigma_1 = 50[\frac{Ns}{m}]$, $c_0 = 10[N]$, $c_1 = 10[\frac{Ns}{m}]$.

2.1.1 *Extension of the elasto-plastic friction model*

2.1.1.1 *Extension for 3D motion*

Since the purpose of the proposed work is to develop a pHRI control algorithm where the robot can move in a 3D space, an extension of the 1D elasto-plastic model in Equation 5 for three dimensions is required. A first solution is to apply such model for each Cartesian axis. While being a simple solution,the behavior of each axis is independent from the others, resulting in a different behavior on each direction and thus to a not smooth motion. In [61] a term λ was proposed to couple forces in the x and y directions for a 2D dynamic tire friction models. On this basis, considering the velocity vector $\dot{\mathbf{x}} = [\dot{x}_x, \dot{x}_y, \dot{x}_z]$, it is possible to rewrite Equation 5 introducing a coupling term $\lambda(\dot{\mathbf{x}})$ equal to:

$$\lambda(\dot{\mathbf{x}}) = \frac{\sqrt{(\mu_{kx}^2 \dot{x}_x)^2 + (\mu_{ky}^2 \dot{x}_y)^2 + (\mu_{kz}^2 \dot{x}_z)^2}}{g(\dot{\mathbf{x}})} \tag{7}$$

where $\mu_{kx}, \mu_{ky}, \mu_{kz}$ are the dynamic friction coefficients along the three axis. For simplicity, the values are chosen such that $\mu_{kx} = \mu_{ky} = \mu_{kz}$. The new three-dimensional model can be expressed as:

$$\begin{cases} F_{d,i} = \sigma_0 z_i + \sigma_1 \dot{z}_i + c_1 \dot{x}_i \\ \dot{z}_i = \dot{x}_i - \gamma_c(\dot{\mathbf{x}}) z_i \end{cases} \quad i = x, y, z \tag{8}$$

Where the function $\gamma_c(\dot{\mathbf{x}})$ is equal to:

$$\gamma_c(\dot{\mathbf{x}}) = \frac{\lambda(\dot{\mathbf{x}})\sigma_0}{\mu_{k,i}^2}, \quad i = x, y, z \tag{9}$$

When two out of three velocities are equal to zero, the model reduces to the one-dimensional form.

2.1.1.2 *Extension for pure elastic movement*

Among the main characteristics of friction, the presliding and stiction are crucial. A friction model demonstrates presliding displacement when a force F_e applied, below a certain threshold value F_{ba}, produce an elastic movement. On the other hand, it demonstrates stiction when the same force does not alter the steady-state value. As mentioned in [62], the LuGre model can render presliding but is unable to demonstrate stiction. This implies that, even with a minimal amount of force, the movement is not entirely elastic. The relative movement x can be represented as the combination of a elastic component z and a the plastic component w. Considering the derivative of this relation, it is possible to define:

$$\dot{x} = \dot{z} + \dot{w} \tag{10}$$

and rearranging the terms:

$$\dot{z} = \dot{x} - \dot{w} \tag{11}$$

which, for $\dot{w} = z\frac{\sigma_0}{\sigma_1}|\dot{x}|$, is the state equation of the LuGre model. The concept of the model expansion suggested in [62] is to maintain the plastic term $\dot{w} = 0$, resulting in $w = 0$. This is achieved by incorporating a new coefficient $\alpha(z, \dot{x})$ into the model, obtaining a new LuGre state equation:

$$\dot{z} = \dot{x} - \alpha(z, \dot{x})|\dot{x}|\frac{\sigma_0}{c_0}z \tag{12}$$

With:

$$\alpha(z, \dot{x}) = \begin{cases} 0 & \text{if } |z| \leqslant z_{ba} \\ \frac{1}{2}\sin(\pi\frac{z-\frac{z_{ss}+z_{ba}}{2}}{z_{ss}-z_{ba}}) + \frac{1}{2} & \text{if } z_{ba} \leqslant |z| < z_{ss} \\ 1 & \text{if } |z| > z_{ss} \end{cases} \tag{13}$$

where z_{ba} and z_{ss} are, respectively, the higher value of z for pure elastic movement and its lower value for plastic movement. For this reason, $0 < z_{ba} < z_{ss}$. A good value for z_{ss} is the one represented in 6. The value of $\alpha(z, \dot{x})$for z within the two limits is an interpolation which makes it a continuous function. Figure 8 presents the response of a single unit of mass to a 2 [N] force applied for 5 [s] and then removed with both base and extended LuGre model. The plot demonstrates that the traditional LuGre model includes a significant plastic component, whereas the advanced extended LuGre model displays a purely elastic movement and it is able to set position to 0 after the force is removed.

2.1.2 *Plastic state reset problems*

As the factor of $\sigma_0 z$ is always present in the model (even during plastic motion) there is always an elastic component present in the movement, which leads to a final elastic discharge at the end of the physical interaction. To achieve a true plastic response, this factor can be multiplied by $(1 - \alpha(z, \dot{x}))$, with $\alpha(z, \dot{x})$ being defined in Equation 13:

$$\begin{cases} F_d = (1 - \alpha(z, \dot{x}))\sigma_0 z + \sigma_1 \dot{z} + c_1 \dot{x} \\ \dot{z} = \dot{x} - \alpha(z, \dot{x})z\frac{\sigma_0}{c_0}|\dot{x}| \end{cases} \tag{14}$$

This extension, while eliminating unwanted elasticity, brings about a critical aspect. Specifically, during plastic motions, $|z| \geqslant z_{ss}$ and the resulting friction force is:

$$F_d = \sigma_1 \dot{z} + c_1 \dot{x} \tag{15}$$

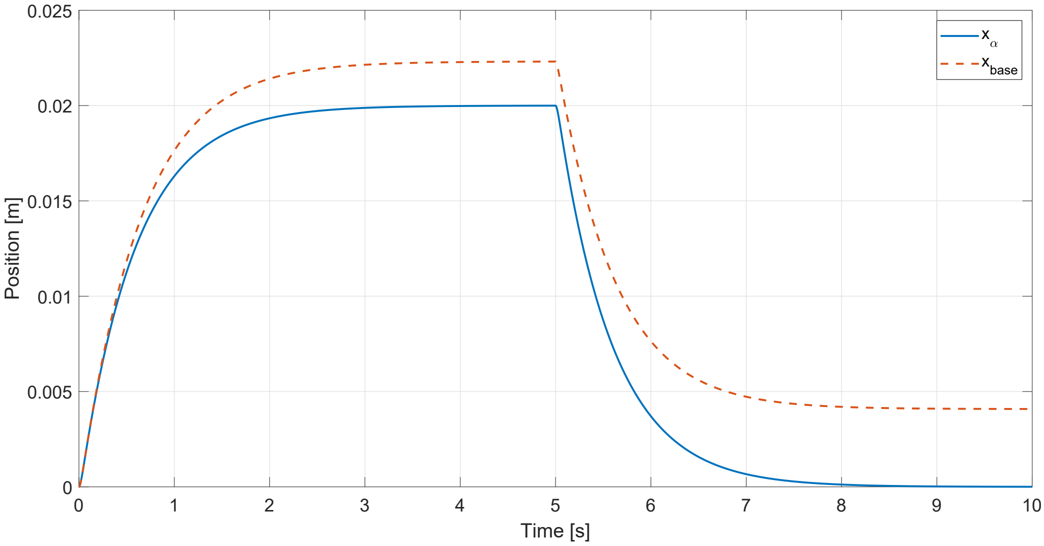


Figure 8: Comparison between the response of LuGre model (represented as a red dashed line) and the response of the extended LuGre model (represented as a blue solid line), with parameters: $\sigma_0 = 100[\frac{N}{m}]$, $\sigma_1 = 50[\frac{Ns}{m}]$, $c_0 = 10[N]$, $c_1 = 10[\frac{Ns}{m}]$, $z_{ba} = 0.9 z_{ss}$.

This implies that there are no forces capable of causing $|z| < z_{ss}$ and thus all subsequent movements will be plastic. As a result, a reset condition must be implemented. Various reset methods have been devised in a one-dimensional and three-dimensional space.

2.1.2.1 *Error-based method*

The reset strategy is based on the error $\varepsilon = \dot{x} - \dot{z}$. In the equation:

$$\dot{z} = \dot{x} - \alpha(z, \dot{x})|\dot{x}|\frac{\sigma_0}{c_0}z \tag{16}$$

the parameter $\alpha(z, \dot{x})$ can be rewritten as:

$$\alpha(z, \dot{x}) = \frac{\varepsilon c_0}{\sigma_0 |\dot{z} - \varepsilon| z} \tag{17}$$

Now, considering the limit for $\varepsilon \to 0$ and solving it:

$$\alpha_0(z, \dot{x}) = \lim_{\varepsilon \to 0} \frac{\varepsilon c_0}{\sigma_0 \dot{z} - \epsilon z} = 0 \tag{18}$$

This result signifies that reducing the error to 0 also reduces $\alpha_0(z, \dot{x})$ to 0. Therefore, an initial reset condition could be to establish $z = 0$ when $\varepsilon = 0$. However, for numerical issues, this outcome cannot be attained. Additionally, with a small $\varepsilon_{limit} \geqslant 0$, if the external force is removed before $\varepsilon \geqslant \varepsilon_{limit}$ the reset does not happen. A possible solution is to reset z when:

$$|\varepsilon(t_n)| \leqslant \alpha(t_n) \max\left(|\dot{z}(t_0), ..., \dot{z}(t_n)|\right), \tag{19}$$

Where t_0 is equal to the time t_n calculates whenever $\alpha(t_n) - \alpha(t_{n-1}) > 0$. By extending this condition in 3D-space the absolute values of the error are replaced with the norms:

$$||\varepsilon(t_n)|| \leqslant \alpha(t_n) \max\left(||\dot{z}(t_0), ..., \dot{z}(t_n)||\right), \tag{20}$$

The trigger threshold is not fixed and it is adjusted to the dynamic of the system. Figure 9 shows an external force applied on the system that produces the values of z and reset signal represented in Figure 10.

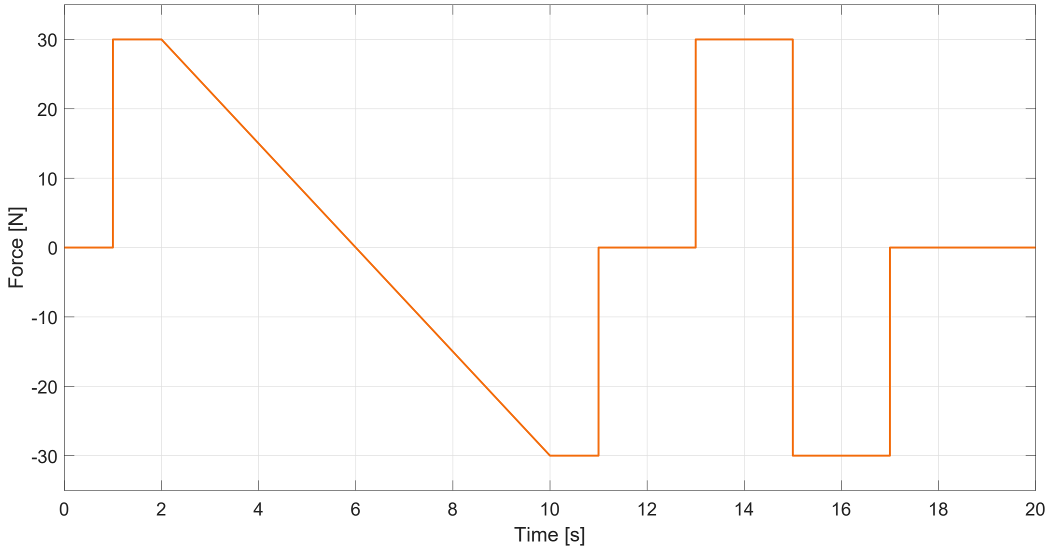


Figure 9: Reset with dynamic error condition method and model parameters: $\sigma_0 = 800[\frac{N}{m}]$, $\sigma_1 = 80[\frac{Ns}{m}]$, $c_0 = 16[N]$, $c_1 = 150[\frac{Ns}{m}]$, $z_{ba} = 0.9 z_{ss}$.

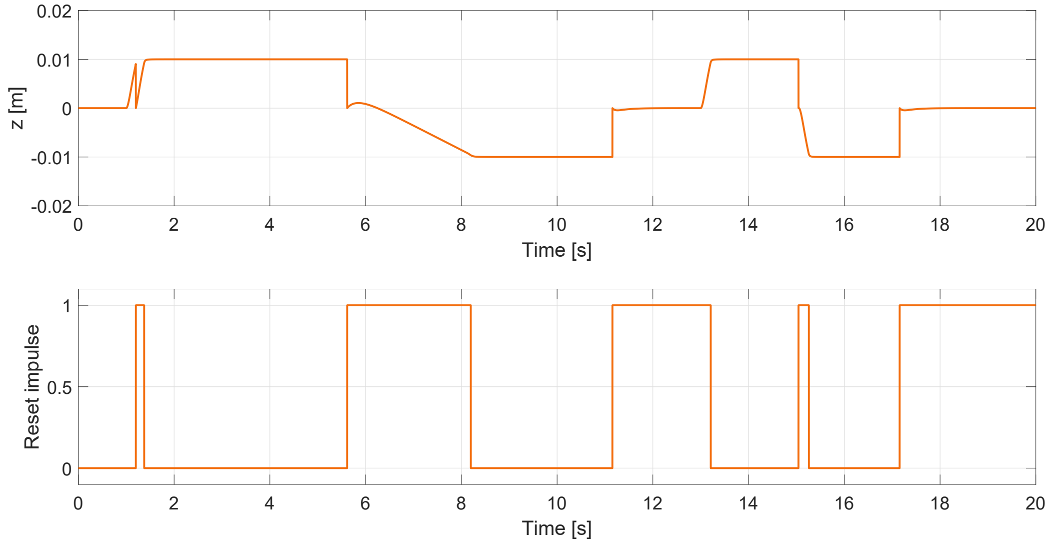


Figure 10: Reset with error-based strategy. fracNm], $\sigma_1 = 80[\frac{Ns}{m}]$, $c_0 = 16[N]$, $c_1 = 150[\frac{Ns}{m}]$, $z_{ba} = 0.9 z_{ss}$.

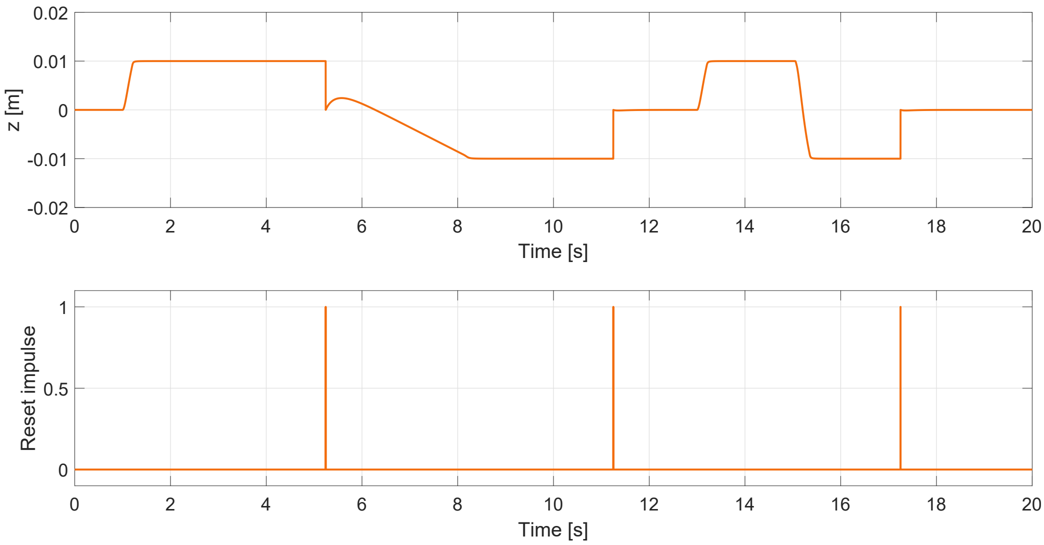


Figure 11: Reset with predictive strategy.$\sigma_0 = 800[\frac{N}{m}]$, $\sigma_1 = 80[\frac{Ns}{m}]$, $c_0 = 16[N]$, $c_1 = 150[\frac{Ns}{m}]$, $z_{ba} = 0.9z_{ss}$.

2.1.3 *Predictive-based method*

This strategy aims to nullify the value of z only when, after a pre-established time t_p from the reset, the forecasted value of z is lower then a specified limit z_{ba}. By identifying t_0 as the instant the condition is examined and t_n as a typical moment within the prediction range, such that $t_0 < t_n \leqslant t_p + t_0$, the computed acceleration is adjusted to $\ddot{x}(tn) = \ddot{x}(t0)$. Subsequently, the prediction is achieved through the equation of uniformly accelerating motion equation

$$\hat{z}(t_p) = z(t_0) + \dot{z}(t_0)t_p + \frac{1}{2}\ddot{z}t_p^2 \tag{21}$$

Since the prediction addresses the scenario following the reset, the value of z at t_0 is equal to 0. Additionally, if the system is in an elastic state, the derivatives of z is: $\dot{Z} = \dot{x}$ and consequently $\ddot{Z} = \ddot{x}$

Therefore, the reset criteria is met if the following condition is satisfied:

$$\left|\dot{x}(t_0)t_p + \frac{1}{2}\ddot{x}t_0 t_p^2\right| < z_{ba} \tag{22}$$

and for three-dimensional extension:

$$\left\|\dot{x}(t_0)t_p + \frac{1}{2}\ddot{x}t_0 t_p^2\right\| < z_{ba} \tag{23}$$

To decrease oscillations after the reset, a second condition was added:

$$\left|\dot{x}(t_0)\frac{t_p}{2} + \frac{1}{2}\ddot{x}t_0(\frac{t_p}{2})^2\right| < z_{ba} \tag{24}$$

and for three-dimensional extension:

$$\left|\left|\dot{x}(t_0)\frac{t_p}{2}+\frac{1}{2}\ddot{x}t_0(\frac{t_p}{2})^2\right|\right| < z_{ba} \tag{25}$$

In this way, to apply the reset condition the Equation 22 and 24 must be true for one dimensional motion. Similar, 23 and 25 must be satisfied for three-dimensional motion. From an external force applied as in Figure 9, the system with predictive reset produces the values of z in Figure 10. This reset method was chosen in the next tests, since it produces the least amount of unnecessary resets .

2.1.4 *The complete model*

Using all the extensions outlined in previous sections, the complete LuGre model for the elasto-plastic controller for 3D-space robot's movement can be defined as:

$$\begin{cases} F_{d,i} = (1-\alpha(\mathbf{z},\dot{\mathbf{x}}))\sigma_0 z_i + \sigma_1 \dot{z}_i + c_1 \dot{x}_i \\ \dot{z}_i = \dot{x}_i - \alpha(\mathbf{z},\dot{\mathbf{x}})\gamma_c(\dot{\mathbf{x}})z_i \\ \gamma_c(\dot{\mathbf{x}}) = \frac{\lambda(\dot{\mathbf{x}})\sigma_0}{\mu_{k,i}^2} \end{cases} \qquad i = x, y, z \tag{26}$$

With:

$$\lambda(\dot{\mathbf{x}}) = \frac{\sqrt{(\mu_{kx}^2\dot{x}_x)^2 + (\mu_{ky}^2\dot{x}_y)^2 + (\mu_{kz}^2\dot{x}_z)^2}}{c_0} \tag{27}$$

and:

$$\alpha(\mathbf{z},\dot{\mathbf{x}}) = \begin{cases} 0,\ \text{if } ||\mathbf{z}|| \leqslant z_{ba} \\ \frac{1}{2}\sin\left(\pi\frac{||\mathbf{z}||-\frac{z_{ss}+z_{ba}}{2}}{z_{ss}-z_{ba}}\right)+\frac{1}{2}\ \text{if } z_{ba} \leqslant ||\mathbf{z}|| < z_{ss} \\ 1,\ \text{if } ||\mathbf{z}|| > z_{ss} \end{cases} \tag{28}$$

with $z_{ss}, z_{ba} > 0$ and $\mu_{kx} = \mu_{ky} = \mu_{kz} > 0$. The type of motion is the same for all three dimensions, thanks to the coupling term $\lambda(\dot{\mathbf{x}})$. Figure 12 illustrates the overall design of the suggested control method.

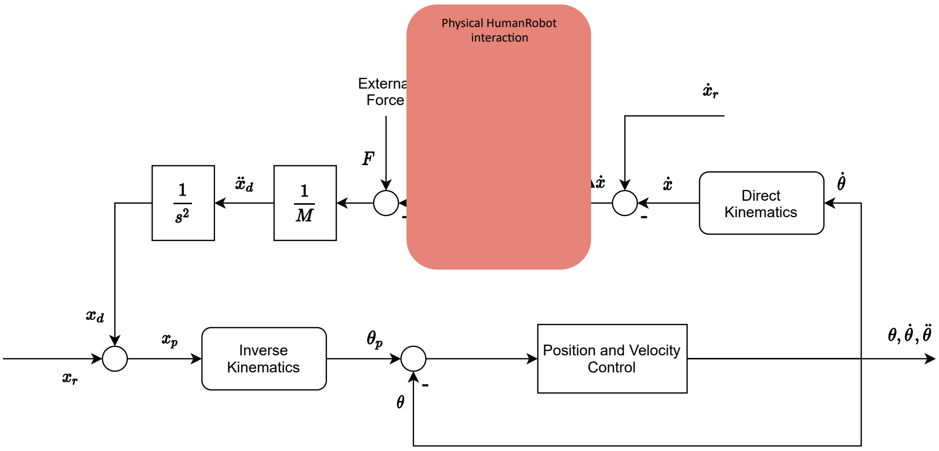


Figure 12: The complete control loop of the proposed Elasto-plastic control.

A simple evaluation of the coupling effect can be derived considering a 2D motion, for simplicity. Let consider an external force F applied on the entire control system in Figure 12 and a mass $\mathbf{M} = \begin{bmatrix} M_x & 0 \\ 0 & M_y \end{bmatrix}$ such that:

$$\mathbf{F} = \begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} F\sin(\omega t) \\ F\cos(\omega t) \end{bmatrix} \forall t \in [0, 10] \tag{29}$$

Using a simulation environment, the evolution of Cartesian positions deformation of the elasto-plastic system $\mathbf{x_d} = \begin{bmatrix} x \\ y \end{bmatrix}$ has been computed. A coupled system will produce a smooth spiral movement along the plane. Figure 13 shows the motion xy of the mass with and without the coupling term $\lambda(\dot{\mathbf{x}})$. As expected, the trajectory results less smooth in the case without coupling.

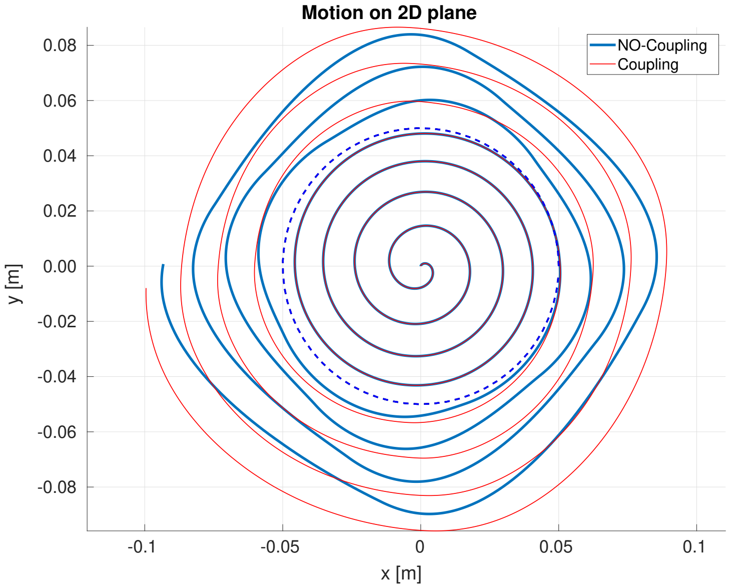


Figure 13: 2D motion of a mass according to the elasto-plastic model presented. The red curve represents the motion for coupled directions of motion and the coupled term $\lambda(\dot{\mathbf{x}})$. The blue curve is the motion while the model is applied separately for each axis. The blue dotted curve is defined by the value of z_{ss}.

The impedance controller receives a Cartesian position vector and generates the virtual friction force necessary to obtain the intended dynamics. To introduce a trajectory reference in a traditional control loop with impedance control, here are two main implementations which can be exemplified using the models in Figure 14.

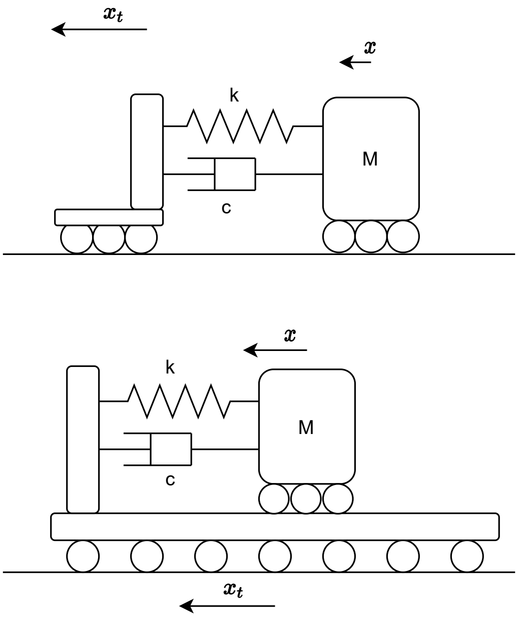


Figure 14: Implementations of trajectory following with impedance control. The second model (bottom) has been used.

Taking into account the second implementation, we have that the dynamic system can be described with the equation:

$$m(\ddot{x} - \ddot{x}_t) + c(\dot{x} - \dot{x}_t) + k(x - x_t) = F_{ext} \tag{30}$$

with the Laplace transformation and rearranging the terms we have:

$$X(s) = X_t(s) + \frac{1}{ms^2 + cs + k} F_{ext}(s) \tag{31}$$

It is easy to see that, without the force component, the mass position follow the trajectory reference perfectly. To reproduce such behaviour and link the information of the position reference x_r with the dynamic of the elasto-plastic system, two operation are required: first, since the robot will be position controlled, the position loop's input must be affected by the dyanmic of the elasto-plast model in such a way:

$$x_p = x_d + x_r \tag{32}$$

Thus, the friction force output F_d (and the dynamic of the elasto-plastic model) directly influences the Cartesian position x_p which regulates the robot's position loop. Second, the elastoplastic controller is fed with the difference $(\dot{x} - \dot{x}_r)$, instead of the single $\dot{x}$ value, linking the damping to the trajectory. The value of x_d can be obtained from Newton's second law of motion, in fact, for each of the three Cartesian axis, an acceleration term is obtained as:

$$\ddot{x}_{d,i} = \frac{F_i - F_{d,i}}{m_i} \qquad i = x, y, z \tag{33}$$

Velocity and position deformations can be obtained by a simple integration. It is important to highlight that m_i is not the mass of robot but instead an inertia parameter which influences the response of the controller.

2.1.5 *Simulation results*

A set of simulations were performed for both one-dimensional and three-dimensional models, with the goal of testing and evaluating the models presented in Equation 26. Figure 15 shows the simulation scheme developed in *Simulink* for three dimension.

A second-order process with a null pole was chosen to simulate the mechatronic system of the robot. To improve the observation of the system's evolution and reduce numerical noise caused by derivatives, the model's integrator was separated to also produce a velocity output, such as:

$$P(s) = \frac{1}{s(7s + 5)} \tag{34}$$

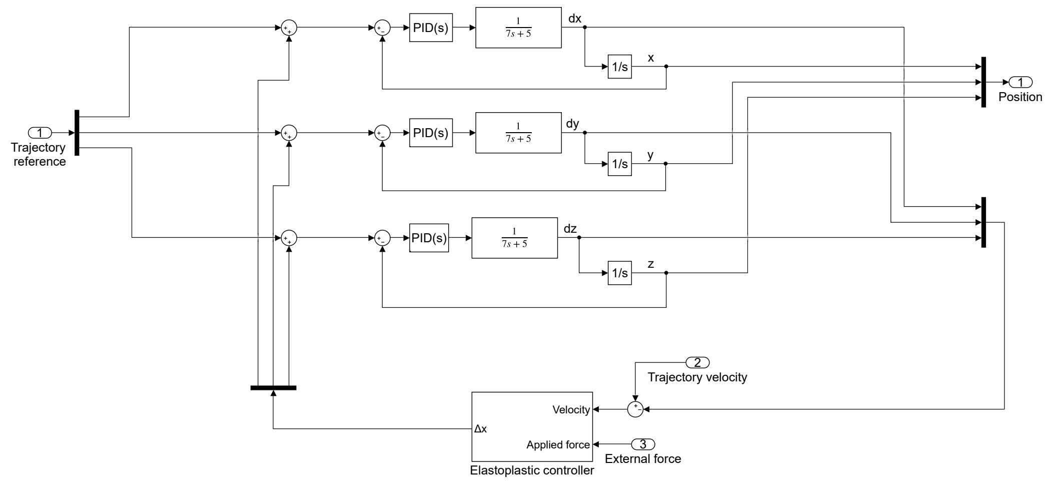


Figure 15: Scheme of the simulation.

To close the robot's loop, a PID controller is been included.

The closed loop was controlled using a PID algorithm with a highly aggressive tuning. The velocity output was fed into the elasto-plastic model and its position output closed the external loop. The control system was used for one dimension and then adapted to three-dimensional space, using three independent control loops for each of the three main Cartesian axes of the robot.

Figure 16 illustrates the behavior of this control system using a force signal obtained from the real robot for both one and thre-dimensional model. Moreover, again in Figure 16 a comparison between the simulated loop and the real robot with the same input force, reference trajectory, and tuning is illustrated.

Through these simulations, a suitable set of parameters were found, then the discretized control was developed in C++ using ROS.

2.1.6 *ROS (Robot Operating System)*

ROS (Robot Operating System) [63, 64] is a comprehensive set of frameworks that facilitates the development and programming of robots through API for the languages C++ and Python. It offers the same functionality as a traditional operating system, but on a cluster of different types of processors. ROS provides a wide range of services, including hardware abstraction, device control through drivers, inter-process communication, application (package) management, and other commonly used functions.

One of the key features of ROS is its ability to represent a set of processes as nodes within a graph. These nodes can receive, send, and multiplex messages from and to other nodes, sensors, and actuators. While responsiveness and low latency are crucial in robot control operations, ROS is

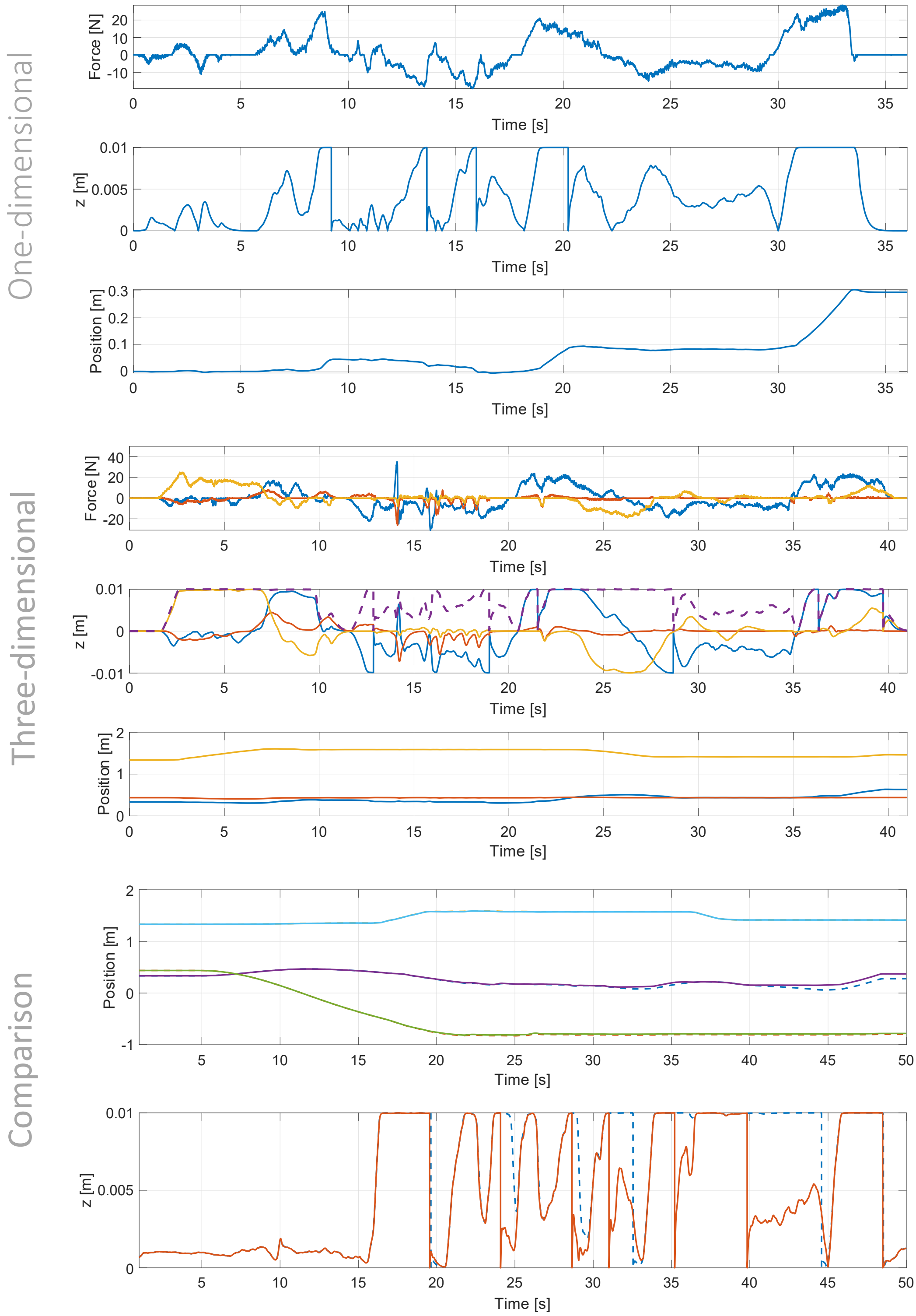


Figure 16: Results of simulation. An input force signal obtained from the robot is applied to the model. For both one and three-dimension the first plot illustrate the force signal. The second plot the state variable z and the third plot the position output. In blue the x axis, in red the y axis, in yellow the z axis and the norm is represented as dashed purple line. The last two plot show a comparison between three dimensional simulated and the real system with the same force profile.

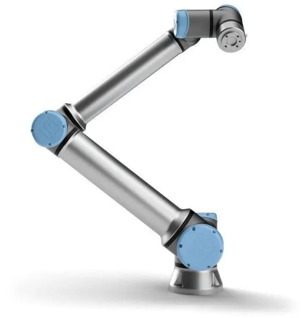


Figure 17: Universal Robots UR10e [1]. A 6-axis robot with 10[kg] payload and a reach of 1300[mm].

Figure 18: Robotiq 2F-85 [2].

not a real-time operating system. However, it is possible to integrate ROS with real-time modules to achieve real-time performance. This makes ROS a powerful tool for developing advanced robotic systems that require efficient and reliable communication between different components.

In summary, ROS is a versatile and robust operating system that provides a wide range of services and features for developing and programming robots. To implement the control it was used the *ros_control* package, which offers two main components: the controller itself and the controller manager. The controller manager acts as a node that manages the controllers' lifecycle. It provides a collection of ROS services to other nodes in the network, allowing them to load and unload controllers at runtime. Additionally, it is responsible for resolving resource conflicts.

2.1.7 *Experimental results*

A set of experiments have been conducted a Universal Robots UR10e (Figure 17). The robot utilized is a 6-axis collaborative robot with a 10kg weight capacity and a reach of 1300[mm]. It was mounted upside down and equipped with a force torque sensor with a 100[N] linear force range in each of its frame axis. The control model defined in previous section was developed in ROS. The cell's base reference was positioned such that the origin was in the middle, the z-axis was vertical and pointing upward, the y-axis was horizontal and running along the width of the cell to the right, and the x-axis was facing the viewer and completing a right frame. The gripper used was a Robotiq 2F-85 (Figure 18), designed specifically for collaborative robots. The control workstation for the cell was running Ubuntu 18.04 and the ROS.

To evaluate the performance and ease of use of the robot with the elasto-plastic control, an experimental collaborative task has been designed, an illustrative scheme is visible in Figure 19.

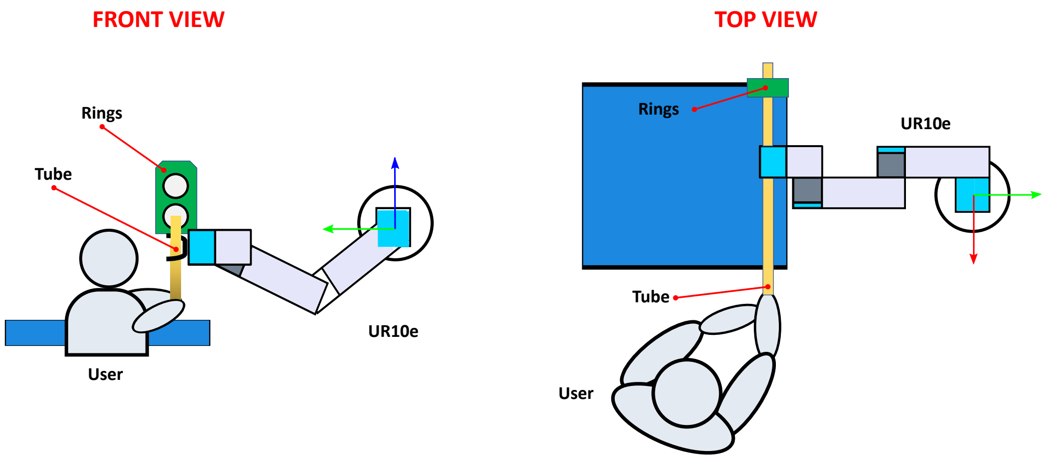


Figure 19: Experimental collaborative task designed to evaluate elasto-plastic control performance.

The objective is to manipulate a cylindrical cardboard tube, measuring $120[mm]$ in length and $56[mm]$ in width. The robot grips one end of the tube while the human holds the other side. In this way a kind of pHRI is achieve by means of the tube. For safety, the robot's capabilities are restricted in terms of the amount of force and speed it can exert. Specifically, the maximum force that can be applied is $120[N]$ and the maximum speed is $750[\frac{mm}{s}]$. Starting from the right side of the cell, the robot follows a path that leads it to the other side of the cell. Mounted on the cell's structure are two rings with a $62[mm]$ radius, one above the other. The participant must exert force on the tube to alter the path and successfully guide the tube through the higher ring and then the lower ring. The path is designed to encourage the participant to move the robot while a reference path is still present. A total of 24 participants took part in the experiments. To gain a deeper insight into the robot's actions, each examination was split into two sections:

- the path monitoring stage, in which the robot's intended path is communicated and the participant must guide it to the initial ring;
- the placement stage, where the actual task is completed.

Each participant was instructed to carry out the task with two different impedance parameters obtained after a pre-tuning session. In fact, during preliminary tests, it was noted that impedance parameters that are optimal for unrestricted movement may not be suitable for precise positioning and vice versa. Moreover, a parameter gain scheduling which is able to swap between these two set has been considered: a the start of the robot it is in "Idle" state (parameters with high impedance), during the trajectory it switch in "Trajectory" state (parameters with low impedance) and for final accurate positioning it swaps again to "Idle" state. Based on the above the subject perform the task three times, with different set of impedance parameters:

- set low impedance "Trajectory". It is helpful for large movements, where the robot has to oppose low resistance and the user is more free to mvoe the robot.
  Parameters: $m = 13[kg]$, $\sigma_0 = 800[\frac{N}{m}]$, $\sigma_1 = 80[\frac{Ns}{m}]$, $c_0 = 16[N]$, $c_1 = 150[\frac{Ns}{m}]$, $z_{ss} = \frac{c_0}{\sigma_0}$, $z_{ba} = 0.9z_{ss}$
- set high impedance "Idle". Helpful for accurate positioning but with limited user movements.
  Parameters: $m = 20[kg]$, $\sigma_0 = 1500[\frac{N}{m}]$, $\sigma_1 = 150[\frac{Ns}{m}]$, $c_0 = 15[N]$, $c_1 = 1280[\frac{Ns}{m}]$, $z_{ss} = \frac{c_0}{\sigma_0}$, $z_{ba} = 0.9z_{ss}$
- Gain scheduling. Automatic shift between "Idle" and "Trajectory". To prevent sudden shifts, transitioning between both sets is done through a linear interpolation.

To evaluate human's satisfaction during the task, each individual assigned a rating, ranging from 1 to 10, to evaluate the performance of the robot during both the trajectory tracking and positioning. A score of 10 indicates that the robot displayed behavior comparable to that of a human operator during the task. The distribution of scores for each phase, separated by tuning, is illustrated in Figure 20 and in Figure 21. The tuning of "Trajectory" received the lowest scores, particularly in the second stage. According to the operators, the main cause of these low scores is

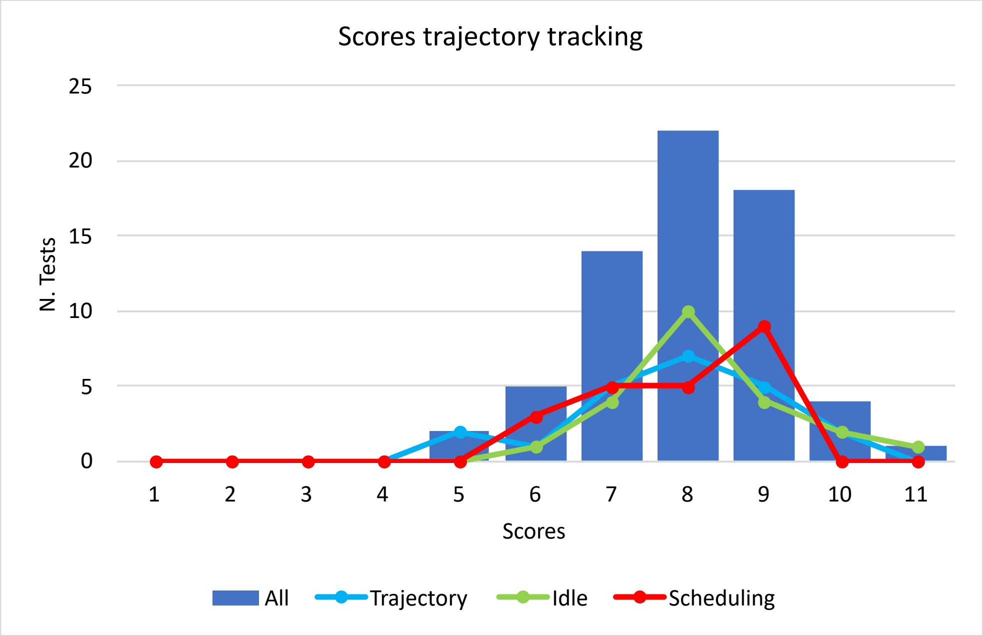


Figure 20: Scores of participants in during trajectoy tracking.

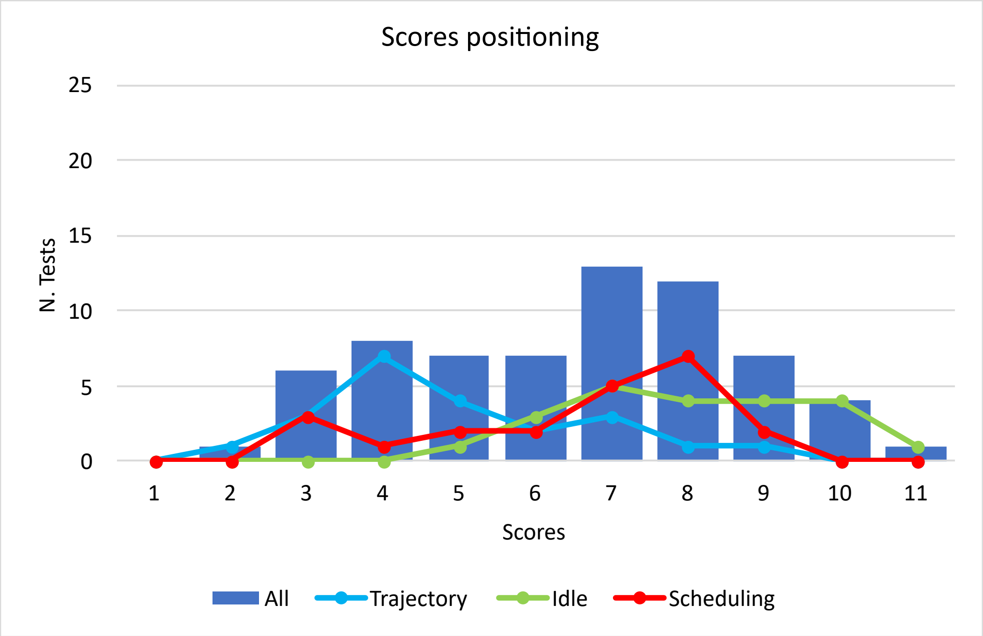


Figure 21: Scores of participants during positioning.

the ongoing elastic reaction. The elastic force opposes to the small forces required for positioning movements. This occurs outside the rings, where positioning is mainly done on the plain orthogonal to the ring axis and during the insertion process, where a force not parallel to the ring axis can cause an elastic burst which leads the robot to instability due to contact forces.

The "Idle" set received high scores for the positioning stage. The greater force opposed by the robot is compensated by the lack of elastic burst, resulting in a more stable system even in the presence of contact forces. Good results were also obtained in the trajectory stage.

The scheduling approach received positive feedback in both stages. Low scores were given only in the second stage, and a possible explanation is an incomplete separation between the two stages, resulting in the tube insertion occurring before the parameter switching is completed.

One other metric to evaluate the type of movement during each set of parameters is the path traveled by the robot (Figure 22). An optimal tuning should enable accurate position phase and prevent the need for subsequent adjustments. In contrast, in case of high levels of elasticity during the positioning stage, this results in a longer overall path, as the robot will push back (due to the spring behavior) after a minimal force input, leading to increased the distance covered. In fact, Figure 22 confirm this: longer path are related to "Trajectory" set and "Scheduling", due to the presence of the "Trajectory" parameters during parameters swap.

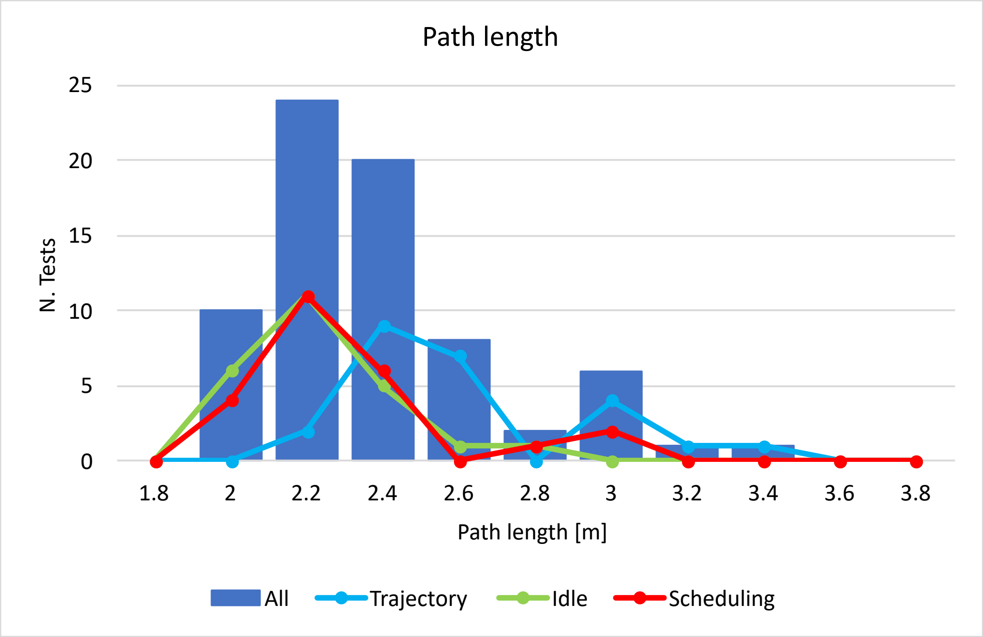


Figure 22: Values of path lengths.

An analysis based on the force data collected has been also carried out, in order to affirm if the controller has a good tuning for pHRI. Figure 23 and Figure 24 show the distribution of average forces applied by each subject both for trajectory and positioning phase. In the "Trajectory" tuning the user need less force, and the overall distribution is small compared with "Idle" tuning. In comparison, scheduling mode shows there is a more consistent pattern with a narrow distribution, particularly during the second phase, indicating noteworthy results.

The total force required during the task is depicted in Figure 25, which displays the integral of the force applied to the robot throughout the entire task. Once more, the scheduling presents a narrow distribution which emphasizes a more steady behavior among diverse participants.

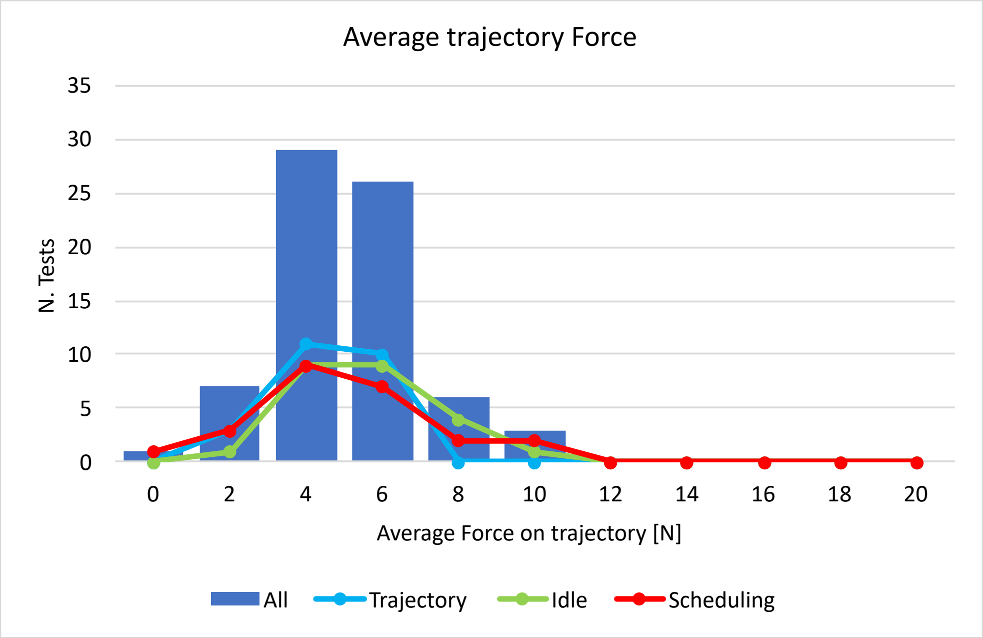


Figure 23: Average trajectory force.

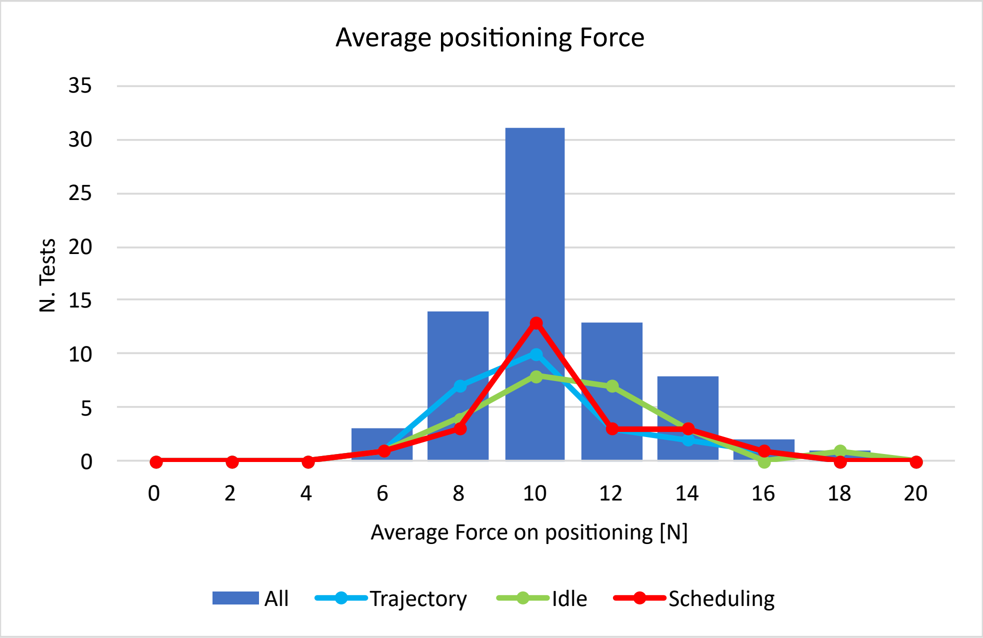


Figure 24: Average positioning force.

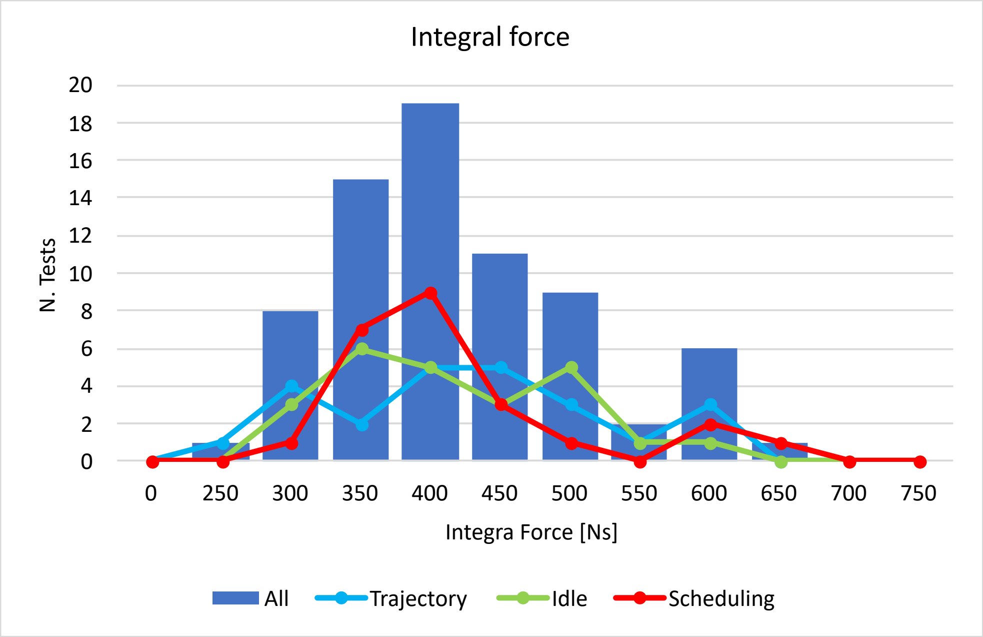


Figure 25: The total force required during the task.

2.2 FUZZY CONTROL BASED ON A HUMAN INTENTION MODEL

As illustrated in the Section 2.1, an elasto-plastic control can lead to useful results when it is necessary to perform a pHRI task in which humans must perform different actions, such as rapid robot movement and precise positioning. Also as demonstrated by the use of dynamically varying impedance parameters and the use of gain scheduling techniques, it was decided to develop a control strategy based on a variable impedance control (VIC) through a model of human intention. Different solutions to manually control a robot (with manual guidance) are strongly related to a precise model of the human arm, a specific knowledge of the task, initial robot learning phase, and many time collects data from electromyogram analysis (EMG) that could be affected by difficulties in processing EMG signals, such as bad signal-to-noise ratio or the fact that the sensor not contact less, which may not be pleasant. From this perspective, it can be evinced how a more simpler HRI model could reduce the level of the problems occurred, if the task fit proper requirements such as, for instance, simplicity of execution. Consider these problems we proposed a control astrategy based on a model of the human-robot chain which does not require a previous identification of the human arm impedance or a detailed knowledge of the task, instead, it is based on a simpler quasi-linear mathematical model to describe the human operator and its activity with a totally offline method, as the one investigated by McRuer [43]. On this basis, a fuzzy-based control strategy with a variable Cartesian impedance control able to adapt online parameters such as mass, stiffness, damping by exploiting a simple HRI model as been developed. The main goal of the developed structure is to speed up industrial tasks that require pHRI, while keeping accuracy.

2.2.1 *HRI with variable impedance control scheme*

When the robot interacts with the environment, a force control schema is widely used to track a force reference and regulate the interaction with surroundings [65, 66]. However, when the manipulator interacts with a partially known environment, setting a correct force reference is not quiet simple. On the contrary, impedance control can be considered a natural evolution of such approach. The proposed control strategy exploits an impedance control strategy for the task space (i.e. Cartesian impedance control). Only Cartesian linear motion is considered; rotational components have been neglected for sake of simplicity, but they can be included to get a more complete model, in which case the procedures to be followed are similar. Impedance Cartesian control, and its complementary admittance control, are considered among the most popular strategies in pHRI, as largely documented and surveyed in literature (e.g. [67]). The aim is to regulate the dynamic characteristics of the impedance to achieve precise goals. Indeed, a variable impedance controller is implemented [68, 69]. A Fuzzy In- ference System (FIS) permits the variation of inner impedance parameters using joint position sensors and an external force sensor, and the tuning of the Fuzzy system has been performed by optimizing a control loop including the McRuers pHRI model. The overall structure is based on previous works on pHRI (e.g. [70, 71]). An illustrative schema is illustrated in Figure 26.

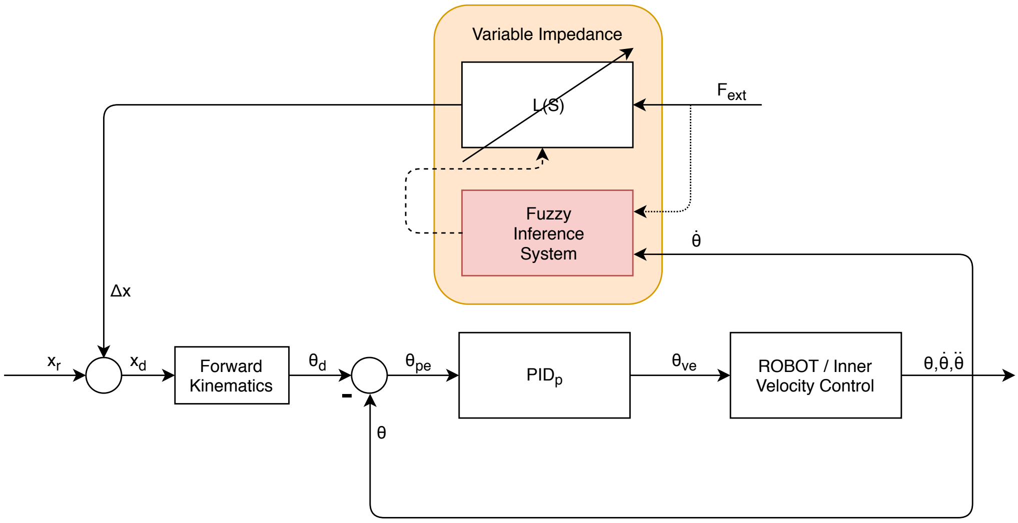


Figure 26: Scheme of Cartesian impedance control. A FIS controller dynamically regulate the impedance parameters online.

A typical impedance control in a 3D task space is based on the following dynamic system [72]:

$$\mathbf{M}_r\Delta\ddot{\mathbf{x}} + \mathbf{D}_r\Delta\dot{\mathbf{x}} + \mathbf{K}_r\Delta\mathbf{x} = \mathbf{F}_{ext} \tag{35}$$

Where $\Delta\mathbf{x}$ is the vector of end-effector translational motion position error, equal to:

$$\Delta\mathbf{x} = \mathbf{x_r} - \mathbf{x} \tag{36}$$

$\mathbf{x}$, and $\mathbf{x_r}$ are, respectively, the robot measured Cartesian position and the related Cartesian target position reference. Similar definitions are made for velocity $\Delta\dot{\mathbf{x}}$, $\dot{\mathbf{x}}$, $\dot{\mathbf{x}}_\mathbf{r}$ and acceleration $\Delta\ddot{\mathbf{x}}$, $\ddot{\mathbf{x}}$, $\ddot{\mathbf{x}}_\mathbf{r}$ values. $\mathbf{M}_r = diag(m_{r1}, ..., m_{r3})$, $\mathbf{D}_r = diag(d_{r1}, ..., d_{r3})$, $\mathbf{K}_r = diag(k_{r1}, ..., k_{r3})$ are the desired impedance matrices composed by translational motion components. $\mathbf{F}_{ext} = [F_{ext,x}, F_{ext,y}, F_{ext,z}]$ is an external force applied by the user and the environment. An inner cascade control loop is implemented: a primary loop regulates the Cartesian position while the secondary regulates the Cartesian velocity. The position reference for the position control loop is $\mathbf{x}_\mathrm{d}$. It can be derived with the following steps. From Equation (35), it can be rewritten as:

$$\mathbf{M}_\mathrm{r}(\mathbf{x_r}-\mathbf{x})s^2+\mathbf{D}_\mathrm{r}(\mathbf{x_r}-\mathbf{x})s+\mathbf{K}_\mathrm{r}(\mathbf{x_r}-\mathbf{x})=\mathbf{F}_\mathrm{ext} \tag{37}$$

In Laplace domain it becomes:

$$\left(\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x_r}=\left(\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x}+\mathbf{F}_\mathrm{ext} \tag{38}$$

The measured position $\mathbf{x}$ is:

$$\mathbf{x}=\mathbf{x_r}-\frac{\mathbf{F}_\mathrm{ext}}{\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}} \tag{39}$$

For the proposed implementation, $\ddot{\mathbf{x}}_\mathbf{r}=0$, thus Equation (38) becomes:

$$\left(\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x_r}=\left(\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x}+\mathbf{F}_\mathrm{ext} \tag{40}$$

$$\mathbf{x}=\frac{\left(\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x_r}-\mathbf{F}_\mathrm{ext}}{\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}} \tag{41}$$

Considering Equation (36), it is possible to define $\mathbf{x_d}$ (the actual position reference) as:

$$\mathbf{x_d}=\mathbf{x_r}+\acute{}\mathbf{x}=\mathbf{x_r}+(\mathbf{x_r}-\mathbf{x}) \tag{42}$$

From Equation (41), the vector $\mathbf{x_d}$ can be defined as:

$$\mathbf{x_d}=2\mathbf{x_r}+\frac{\mathbf{F}_\mathrm{ext}-\left(\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}\right)\mathbf{x_r}}{\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}} \tag{43}$$

Eventually, the new reference for the inner Cartesian position control loop is:

$$\mathbf{x_d}=\frac{2\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}}{\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}}\mathbf{x_r}+\frac{1}{\mathbf{M}_\mathrm{r}s^2+\mathbf{D}_\mathrm{r}s+\mathbf{K}_\mathrm{r}}\mathbf{F}_\mathrm{ext} \tag{44}$$

2.2.2 *Quasi-linear crossover model of the human operator*

When dealing with pHRI that exploits variable impedance control strategies, tasks such as pick&place are reduced to mutual movement between human and robot; in this case the first one is able to deviate from robot's initial trajectory and impose a different one. In such a context, the human operator will act as human motor controller by internally deciding the goal trajectory to perform, and imposing an external force on the Cartesian impedance model to achieve such objective. From this point of view, the performance of the human operator can be well approximated as the action of a inanimate controller, as recently reminded in [73]. This situation results in a simple compensatory manual control system. A base example of such modeling of HRI, has been recently treated also in [30].

One of the most common and simple example of a human control model is the McRuer's Crossover (CO) model [74, 75]. The CO model hints how humans adapt to different plants to elicit stable and effective control responses. Initially, as recently reminded in [76], McRuer deduced a model

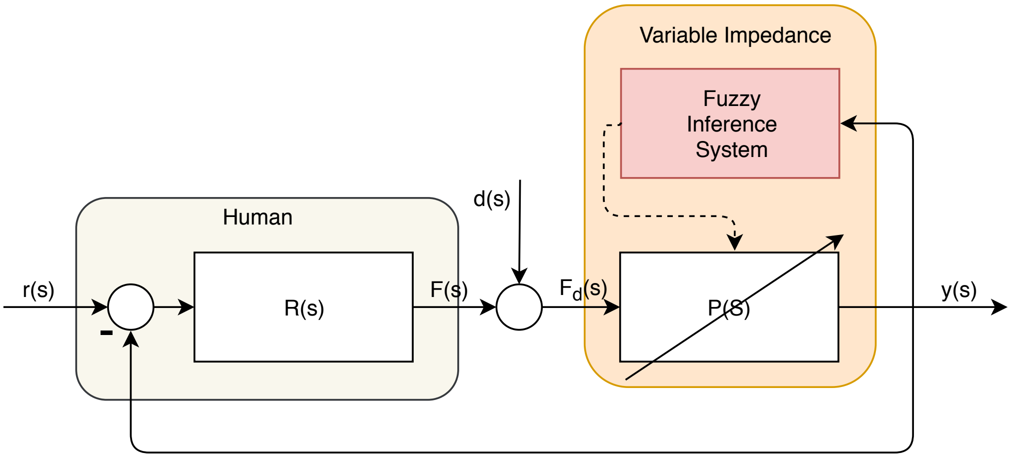


Figure 27: Schema of a pHRI using McRuer's based model. $r(s)$ is the internally decided Cartesian position goal trajectory, $F(s)$ is the force exerted by the user and $F_d(s)$ is the same force affected by an external disturbance $d(s)$.

by recording responses of human subjects to visually presented stimuli in single-loop tracking tasks [77, 78]. In the experiments, the subjects were requested to control a moving element, where the position of the moving element represented an error. Human subjects minimized the error by moving the controlled element towards a reference. It was observed that the manipulative control actions of the human operator are different for different transfer functions, but the combined human-machine behavior is approximately invariant.
It can be convinced, that the CO model exhibits a behavioral invariant of the human in its effort to adapt to the plant, offering a consistent human-machine interaction. The related block diagram can be described like a simple compensatory manual control system as illustrated in Figure 27.

Such control scheme has been taken into account to identify a specific impedance scaling rule that satisfy given goals. Such rule is identified by an optimization algorithm considering only a linear motion along the X axis (for simplify calculi). Thus, consider Equation (35), it becomes:

$$m_1 \Delta\ddot{x} + d_1 \Delta\dot{x} + k_1 \Delta x = F_{ext} \tag{45}$$

The CO model in Figure 27 can be considered as a variation of the proposed structure (Figure 26) focused on the human-robot chain. In this control structure, $r(s)$ is the Cartesian position goal trajectory internally decided by the user that exerts a force $F(s)$. Due by its simplicity, the model proposed aims to avoid common problems related to higher complexity systems.

On the design of the control schema, the inner position loop in Figure 27 is considered perfectly regulated by an adequately tuned PID. The open loop transfer function is defined as:

$$L(s) = R(s)P(s) = \frac{K_c e^{-\tau_c s}}{s} \tag{46}$$

Where magnitude and phase of $L(s)$ are:

$$\begin{aligned} |L(jw)|_{dB} &= 20(\log\omega_c - \log\omega) \\ \phi(L(jw)) &= -\frac{\pi}{2} - \tau\omega \end{aligned} \tag{47}$$

$R(s)$ defines the human operator dynamic, described as a controller that tailor itself to the plant $P(s)$ such that the loop transfer function $R(s)P(S)$ behave as an integrator with a certain delay τ_c at the certain gain crossover frequency ω_c determined by the human operator.

A FIS control will produce an output value K_f to scale-up all impedance parameters at the same time. In this way, it possible to preserve the same value of damping factor ξ and natural frequency ω_n. According to the proposed impedance control scheme (simplified by using only the X axis), $P(s)$ is the impedance model proposed in Equation (45). Thus, the plant transfer function $P(s)$ will become:

$$P(s|K_f) = \frac{y(s)}{F_d(s)} = \frac{\Delta x(s)}{F_d(s)} = \frac{1}{(1-K_f)(m_1 s^2 + d_1 s + k_1)} \tag{48}$$

To preserve the structure defined in Equation (46), $R(s)$ becomes:

$$R(s|K_f) = \frac{L(s)}{P(s|K_f)(T_f s + 1)} = \frac{K_r(m_1 s^2 + d_1 s + k_1)e^{-\tau_c s}}{s(T_f s + 1)} \tag{49}$$

with $K_r = K_c(1-K_f)$ and τ_f is the time constant of a high-frequency filter which make $R(s)$ proper. In this condition, the operator (which acts as a controller) will perfectly track the desired reference on the entire bandwidth while adapting to the dynamic of the plant which changes according to the variable impedance criteria. This change is linear, ideally granting large impedance parameters for low velocities (absorbing disturbances), and smaller ones for high velocities (motion smoothness for the operator). It is supposed that in ideal condition the operator is able to perfectly adapt itself to the plant. However, once an external load disturbance $d(s)$ is introduced, the operator could be no longer be able to track perfectly the reference. The idea is to find a correct scaling rate K_f of impedance parameters under this condition, in order to fulfill the best tracking performance at low velocities and a good smoothness/force reduction for higher ones. Such situation is recreated in the simulation tests.

2.2.3 *Formalization of the fuzzy-based optimization algorithm*

As said Section 2.2.2, the FIS produces an output value K_f to scale-up all impedance parameters. A Takagi Sugeno (TS) method is used. Indeed, the output variable K_f is fixed to feasible, likely functions, while the input-output surface is modeled by varying only the inputs Membership Functions (MFs) parameters, i.e. the values of K_f depends on the FIS rules imposed and the related MFs. An optimization algorithm was designed to find offline the best input MFs parameters which optimize task performance for low velocities (i.e. small position error), and motion smoothness for higher velocities. Two different FIS control strategy are considered: in the first one, only the measured Cartesian velocity is used as a single fuzzy input, then a second additional input has been tested considering the force applied by the user. Both strategies are evaluated in order to identify any performance improvements related to a second fuzzy input. In fact, if the force signal is highly affected by noise, it could result in rapid variation of K_f and thus, impedance parameters, leading the system to manifest unwanted behaviors, like rapid variations from a stiffer state to a more compliant one. In that case, a velocity based FIS control is preferable.

2.2.4 *MATLAB simulation and optimization*

The schema in Figure 27 has been implemented in *Matlab* environment. In order to cover a wide range of Cartesian velocity and applied force values, the simulation is run multiple time using a sinusoidal sweep reference $r(s)$, and we simulate an external disturbance $d_i(s)$ that the system will try to compensate in order to track the reference, according to the considerations specified in Section 2.2.2. Different step disturbances $d_i(s)$ are applied at each run:

$$d_i(s) = \mathcal{L}\{K_i\} \mid K_i = K_a + i\frac{K_b - K_a}{N-1} \ \forall i \in [0, N_d - 1] \tag{50}$$

Where N_d is the number of disturbances, K_i the related $i-th$ gain, and K_a,K_b are, respectively, the minimum and maximum gain. The Cartesian position reference $r(s)$ is equal to:

$$r(s) = \mathcal{L}\left\{D\sin\left(\alpha t^2 + \phi\right)\right\} \ \forall t \in [0, t_{max}] \tag{51}$$

Where D is the amplitude of the movement, α is the angular acceleration and ϕ the sine phase, t_{max} is the maximum time of movement.

Ideally this situation resembles a pHRI task, where human applies a force $F(s)$ and an external disturbance $d(s)$ (such as an external environmental force) acts on the system to infer in the human tailoring with the plant, producing an overall force $F_d(s)$. In the following, $t_{max} = 60[s]$, $D = 1[m]$, $N_d = 3$, $K_a = 0.1$, $K_b = 30$. To give a clarifying example, $K_b = 30$ can resemble a 30[N] disturbance applied on a single Cartesian axis. Different disturbance ranges can be chosen depending on the task. Referring to Equation (48), initial values of mass, spring, damping are chosen equal to: $m_1 = 30[kg]$ $c_1 = 105[\frac{Ns}{m}]$ $k_1 = 120[\frac{N}{m}]$.

Three Gaussian MFs have been defined: *low* $[\mu_1(t)]$, *medium* $[\mu_2(t)]$, *high* $[\mu_3(t)]$. Where t is the input of the MF, it can be a velocity input or a force input. Similarly, three output functions have been chosen with similar names. Each Gaussian MF is defined by:

$$\mu_n(t|\nu,\sigma) = \exp\left(-\frac{(t-\nu_n)^2}{2\sigma_n^2}\right), \forall n \in \{1,2,3\} \tag{52}$$

with ν_n being the center and σ_n^2 the variance of the Gaussian curves. The free variables used for the optimization algorithm are the centers and variances of input Gaussian MFs, defined by the following set:

$$M_p = \{\nu_n, \sigma_n | \; n \in \mathbb{N}, 1 \leqslant n \leqslant 3\} \tag{53}$$

The algorithm runs for N_d times, one for each disturbance applied. For each run, the output position error and the force exerted by the user are saved. Eventually, the absolute position error (APE) and the absolute force (AF) among each test, are calculated. Consider a simulation sample time $s_t = 1 \times 10^{-2}[s]$ and $N = \frac{t_{max}}{s_t} = 6 \times 10^3$ as the number of time iterations. Let $\Delta\mathbf{x_M} \in {}^{N_d \times N}$ and $\mathbf{F_M} \in {}^{N_d \times N}$ be some matrices representing APE and AF respectively, let $\acute{}\overline{\mathbf{x}},\overline{\mathbf{F}} \in {}^{N}$ denote the corresponding vectors of means of the columns and $i, j \in \mathbb{N}$ such that:

$$\Delta\hat{\mathbf{x}} = \left(\frac{1}{N_d}\sum_{i=1}^{N_d}\Delta x_{i,j}, ..., \frac{1}{N_d}\sum_{i=1}^{N_d}\Delta x_{i,N}\right) \tag{54}$$

$$\hat{\mathbf{F}} = \left(\frac{1}{N_d}\sum_{i=1}^{N_d}F_{i,j}, ..., \frac{1}{N_d}\sum_{j=1}^{N_d}F_{i,N}\right) \tag{55}$$

The aim of the optimization algorithm is to find the parameters of the set defined in Equation (53) by minimizing the following cost:

$$\mathrm{cost} = \sum_{i=1}^{N_t} \frac{\gamma_i^{\lambda_1}\Delta x_i}{\max(\hat{}\mathbf{x})} + \frac{(1-\gamma_i)^{\lambda_2} F_i}{\max(\hat{\mathbf{F}})}$$

$$\text{such that:} \begin{cases} K_f \leqslant 0.1, & \mathit{with}\ \dot{\hat{x}} = 0 \\ K_f \geqslant 0.85, & \mathit{with}\ \dot{\hat{x}} = 1 \\ -10 \leqslant \nu_n \leqslant 10 \\ 0.01 \leqslant \sigma_n \leqslant 1 \end{cases} \qquad (56)$$

Where γ_i balances the two components during time. In particular, it maximize Δx_i (APE) at low velocities and minimize it for higher ones. On the contrary, it produces the opposite effect for F_i (AF). It is defined as:

$$\gamma_i = \frac{i}{N_r - 1} \qquad (57)$$

Indeed, generally it is preferable to reduce the position error for low velocities, since in collaborative pHRI tasks in which the human moves the robot (manual-guidance), high-precision tasks are often performed at low speeds to have greater trajectory control. On the contrary, for greater speeds, high precision is no longer required, instead, reducing the total force exerted by the user and increase task smoothness is more recommended. Furthermore, λ_1 and λ_2 weight both components to achieve the desired trade-off between precision and smoothness.

A standard laptop equipped with an Intel i7-7700HQ CPU @ 2.80GHz x8 and 16GB of RAM is used in Matlab environment. Since the optimization problem is not convex, it was considered a multistart algorithm with 11 initial conditions (10 random points and 1 additional condition equals to MFs parameters of the initial linear input-output surface). A step tolerance equal to 1×10^{-3} is considered.

As said in Section 2.2.3, two different FIS control strategy are considered. For the first one, a single Cartesian velocity input is applied. In the second one, an additional input based on the user's applied force F_{ext} is applied.

2.2.5 *Single fuzzy input control - design and simulation results*

The first FIS control strategy consider the scaled norm of the measured robot Cartesian velocities $\dot{\mathbf{x}}$ as fuzzy input:

$$\dot{\hat{x}} = \frac{||\dot{\mathbf{x}}||}{\dot{x}_{max}} \qquad (58)$$

Where $\dot{x}_{max}$ is the maximum Cartesian velocity of the robot manually moved by the operator. It is set equal to $\dot{x}_{max} = 0.5[\frac{m}{s}]$. Thus, in Equation (52) $t = \dot{x}$. The value of $\dot{x}$ is used to discern the correct K_f, adapting online the impedance parameters.
The following fuzzy rules have been defined:

$$\begin{cases} \#1 \text{ IF } \dot{x} : \mu_{\dot{x},1}(\dot{x}) \text{ THEN } K_f & = a_1\, \dot{x} + b_1 \\ \#2 \text{ IF } \dot{x} : \mu_{\dot{x},2}(\dot{x}) \text{ THEN } K_f & = a_2\, \dot{x} + b_2 \\ \#3 \text{ IF } \dot{x} : \mu_{\dot{x},3}(\dot{x}) \text{ THEN } K_f & = a_3\, \dot{x} + b_3 \end{cases} \tag{59}$$

Where, $b_1 = b_2 = b_3 = 0$ and $a_1 = 0.05, a_2 = 0.5, a_3 = 0.95$. It can be evinced that constant functions are used to model the output functions. Considering the K_f variation in Equation 48, the rule #1 means high impedance parameters; the human aims to move the manipulator at low velocities. High impedance levels produce higher precision in task execution and good rejections of disturbances. Rule #2 is a compromise between low and high levels of impedance. Rule #3 mean less resistance induced by the plant and a fast, smooth, low-demanding movement for the operator.

Based on the such assumptions, the optimization algorithm previously described found suitable values of impedance parameters. The algorithm is run using the MF parameters of a linear scaling function as starting points. Such function is obtained by means of the MFs, illustrated in Figure 28. The linear case will be frequently compared to the proposed FIS to evaluate the performance of the controller. In fact, the linear case can be seen as a valid benchmark, being in fact the simplest case for implementing variable impedance with Fuzzy control.

The algorithm was run three times at three different preset of λ in order to analyze the impact of very different λ values on Δx_i and F_i in Equation (56), identify a possible λ candidate that improve performance. The algorithm converged for all sets of λ considered, identifying the MFs parameters σ_n and ν_n observable in Table 1. The related input-output surfaces are illustrated in Figure 29. λ_1 and λ_2 can be generally modulated to shape the input-output as desired. By modify both λ, it is possible to achieve a high level of freedom in terms of balancing the cost function. However, to avoid complexity, it has been decided to fix the weight $\lambda_2 = 2.5$ and let the weight trade-off be managed by λ_1 alone. Hence, for each i-th test, λ_1 has the following values:

$$\lambda_{1,i+1} = 5\lambda_{1,i} \;\forall i \;\in [0, 2] \tag{60}$$

With $\lambda_{1,0} = 0.1$. In this conditions, the more λ_1 increases the more K_f function is moved on the left side of the plane in Figure 29. In fact, increasing λ_1 means reduce the value of $\gamma_i^{\lambda_1}$ in Equation 56, which reduces the

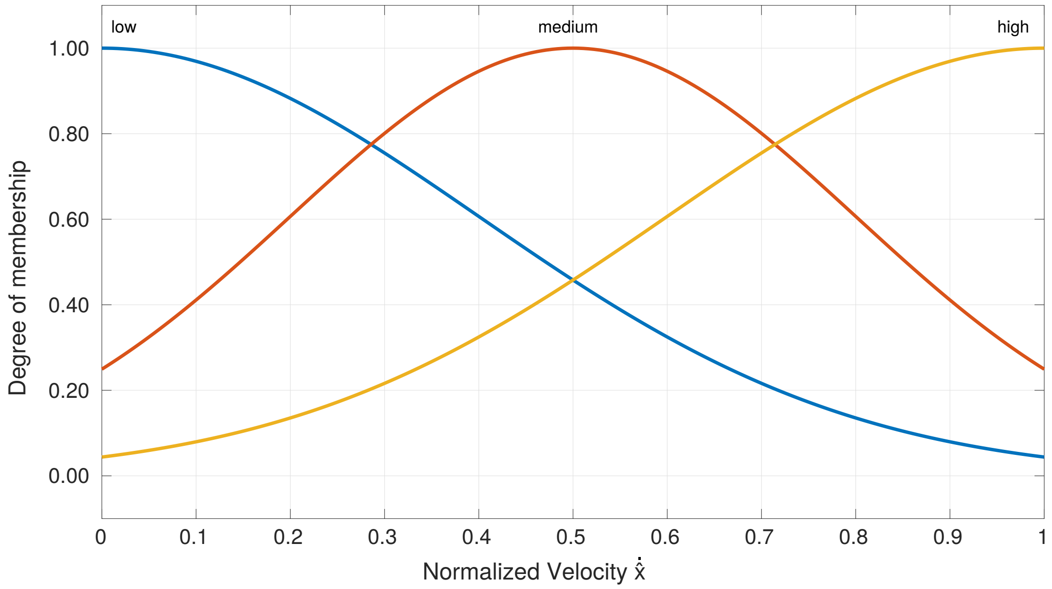


Figure 28: Initial membership functions related to a fuzzy input-output surface for linear scaling: $K_f(t) = t$, with t as a fuzzy input. Used as starting point for the optimization algorithm. In blue, the *'low'* MF. In red, the *'medium'* MF. In yellow, the *'high'* MF.

Table 1: Input MFs parameters identified by the optimization algorithm. Three different sets of λ have been used.

Test	λ_1	λ_2	M_p	MFs		
				1	2	3
1	0.100	2.50	σ_1	0.178	0.344	0.473
			ν_1	0.282	-2.59	-1.63
2	0.500	2.50	σ_2	0.387	0.351	0.581
			ν_2	-1.74	4.82	-2.30
3	2.50	2.50	σ_3	0.533	0.636	0.801
			ν_3	1.53	-5.15	3.412

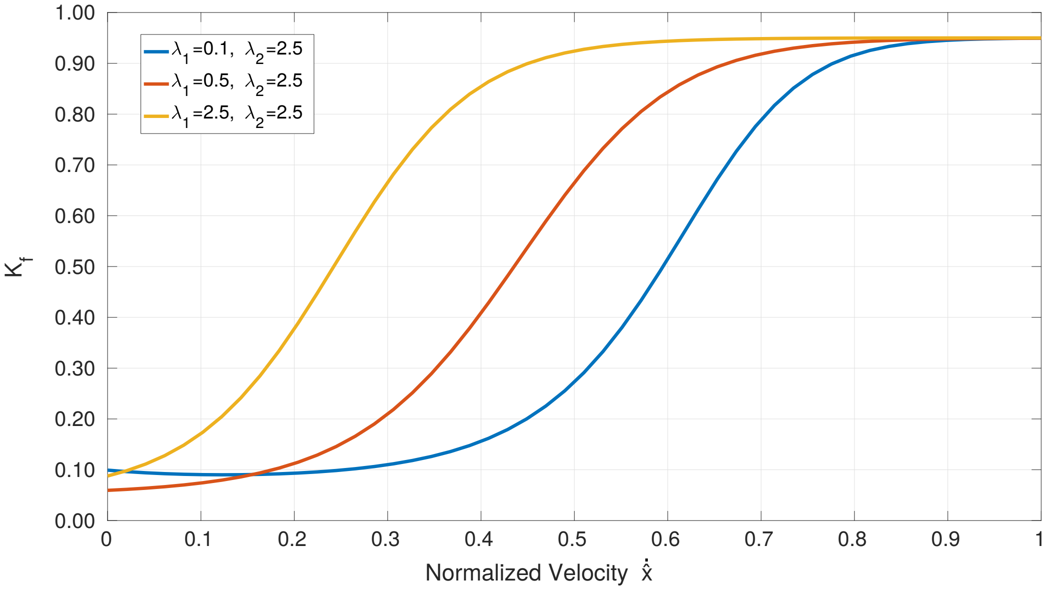


Figure 29: Fuzzy input-output surface for different λ tested. Blue line: $\lambda_1 = 0.1, \lambda_2 = 2.5$, red line: $\lambda_1 = 0.5, \lambda_2 = 2.5$, yellow line: $\lambda_1 = 2.5, \lambda_2 = 2.5$

.

Table 2: Main figures of merit considered.

Test	FOM	d_i			$FOM_{\sigma\%}$
		1	2	3	
1	ITAEP	2.89×10^5	$2.89{\times}10^5$	2.88×10^5	0.20%
	ITAF	1.11×10^7	$1.13{\times}10^7$	1.20×10^7	4.12%
2	ITAEP	2.89×10^5	2.88×10^5	2.86×10^5	0,53%
	ITAF	0.913×10^7	0.937×10^7	1.08×10^7	9.24%
3	ITAEP	2.89×10^5	2.86×10^5	2.82×10^5	1.23%
	ITAF	0.598×10^7	0.651×10^7	0.935×10^7	24.9%

importance of Δx_i (APE) and increase more F_i (AF). Indeed, as illustrated in Figure 29, by rewarding more F_i (AF) the scaling becomes higher also at low velocities, means lower force values as control output also during small velocities.

2.2.5.1 *Simulation results*

For each simulation performed, two figures of merit (FOM) have been considered, both related to the integral of the absolute error multiplied by time (ITAE). The first one rewards more the Cartesian position error for lower values of time, the second one reward more the user's applied force for higher values of time. Each FOM is named, respectively, ITAEP (*p* for position) and ITAF (*F* for force). Both value are illustrated in Table 2.

Moreover, standard deviation relative to the mean of FOM ($\mathrm{FOM}_{\sigma\%}$) is considered. Looking at Table 2 and considering $\mathrm{FOM}_{\sigma\%}$, it is possible to evince how different λ at various disturbances d_i produce a limited variation of ITAEP. This suggests that the control algorithm is robust enough to different values of disturbance; considering only variation on robot's Cartesian position this is limited. On the contrary, ITAF variation is significant among all different tests and disturbances considered, in particular Test 3 has a high deviation. A ITAF comparison between FIS controller using Test 3 and the linear FIS among $N_d = 3$ values of disturbances is represented in Figure 30. More precisely: when d_1 is applied a 33.9% ITAF reduction is observable, for d_2 a 24.8% reduction, and for d_3 a 9.75% reduction. The values of the integrate time absolute force exerted during the simulation trials results generally reduced, when compared with the linear case.

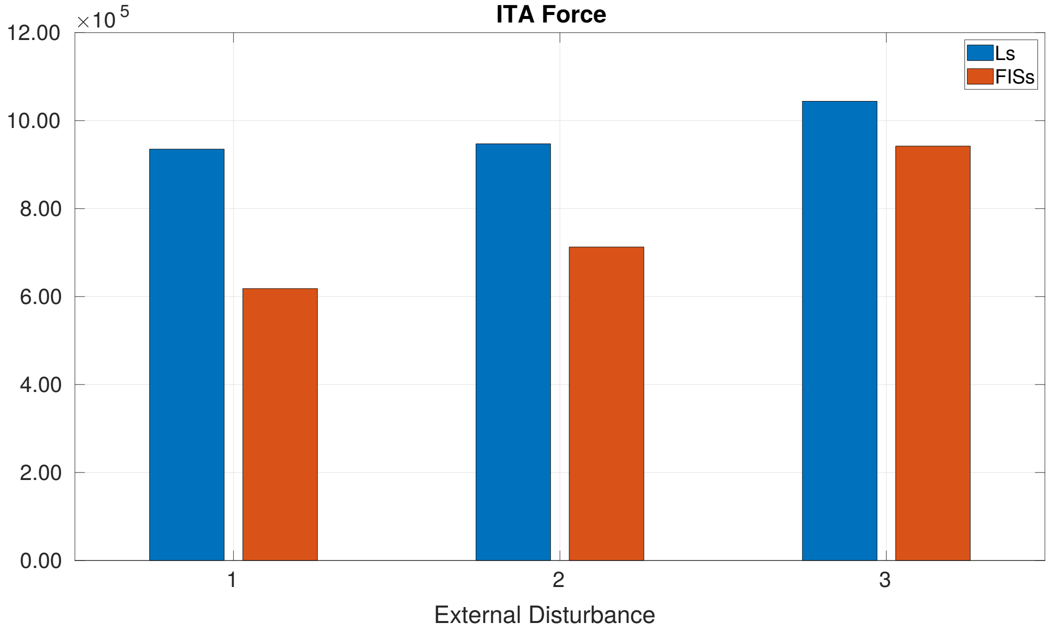


Figure 30: A ITAF comparison between Test 3 FIS controller and the linear FIS controller with $N_d = 3$ disturbances applied. The blue bars represent the linear fuzzy scaling, the red bars represent the optimized FIS velocity scaling controller obtained. It can be evinced how the force required by the controller (human) is reduced in the second case.

2.2.6 *Two fuzzy input control - design and simulation results*

Here we consider a FIS with a force contribution as additional input. In the case the force sensor is not affected by low signal-to-noise ratio, resulting in rapid variation of K_f that could lead the system to an unstable behavior, an additional fuzzy input can be implemented es an extension of the previous strategy, analyzing all the possible improvements. The velocity input MFs are the same calculated by the optimization algorithm in the previous subsection. The force MFs parameters are initially chosen so that they recreate an input-output surface equal to the one in the Figure 29, in which the force contribution is initially irrelevant (in this case it will be a 3-D plot, since we have two independent variables: velocity and force).

Then, we will try to identify the best input-output fuzzy surface by varying only one MF, so as to provide a degree of freedom to the system without using more complex optimization algorithms. The goal will be to identify the best surface that minimizes FOMs related to APE and AF, by simulating and performing different tests with the McRuer model in Figure 27, considering multiple external disturbances for each test. Each MF parameter is reported in Table 3.

Table 3: MFs parameters of the second fuzzy input, related to scaled norm of the force applied by the user $\hat{F}_{ext}$.

Mp	MFs		
	1	2	3
σ	0.177	0.177	0.177
ν	0.0	0.5	1.0

Hence, additional FIS rules related to force have been considered:

$$\begin{cases} \#1 \text{ IF } \dot{x} : \mu_{\dot{x},1}(\dot{x}) \text{ THEN } K_f & = a_1\, \dot{x} + b_1 \\ \#2 \text{ IF } \dot{x} : \mu_{\dot{x},2}(\dot{x}) \text{ THEN } K_f & = a_2\, \dot{x} + b_2 \\ \#3 \text{ IF } \dot{x} : \mu_{\dot{x},3}(\dot{x}) \text{ THEN } K_f & = a_3\, \dot{x} + b_3 \\ \#4 \text{ IF } \dot{x} : \mu_{\dot{x},1}(\dot{x}) \;\&\&\; \hat{F}_{ext} : \mu_{\hat{F}_{ext},1}(\dot{x}) \text{ THEN } K_f & = a_1 g(\dot{x}, \hat{F}_{ext}) + b_1 \\ \#5 \text{ IF } \dot{x} : \mu_{\dot{x},3}(\dot{x}) \;\&\&\; \hat{F}_{ext} : \mu_{\hat{F}_{ext},3}(\dot{x}) \text{ THEN } K_f & = a_3 g(\dot{x}, \hat{F}_{ext}) + b_3 \end{cases} \tag{61}$$

Where $\hat{F}_{ext}$ is the scaled norm of the force applied by the user:

$$\hat{F}_{ext} = \frac{||\mathbf{F}_{ext}||}{F_{ext,max}} \tag{62}$$

The variable $F_{ext,max}$ is the maximum force exerted by the user set equal to $100[N]$. As it has been said, the FIS controller has been simulated multiple times using the McRuer model in Figure 27 considering multiple external disturbances for each test. The value N_d in Equation (50) is set to 15. The MF that we decide to vary in order to figure out which fuzzy input-output surface shape to use is: $\mu_{\hat{F}_{ext},3}$. In particular, only the center ν of MF $\mu_{\hat{F}_{ext},3}$ is varied N times. It results in:

$$\nu_i = \nu_a + \frac{\nu_b - \nu_a}{N-1} i \;\forall i \in [0, N-1] \tag{63}$$

Where $\nu_b = 1.1$ and $\nu_a = 0.6$. As an example, the FIS surface at $\nu_i = \nu_b = 1.1$ is represented in Figure 31. By reducing ν_i towards 0.6 the resulting high values of K_f (i.e. the yellow surface in Figure 31) will move the parameter $(1 - K_f)$ in Equation 48 close to 0.

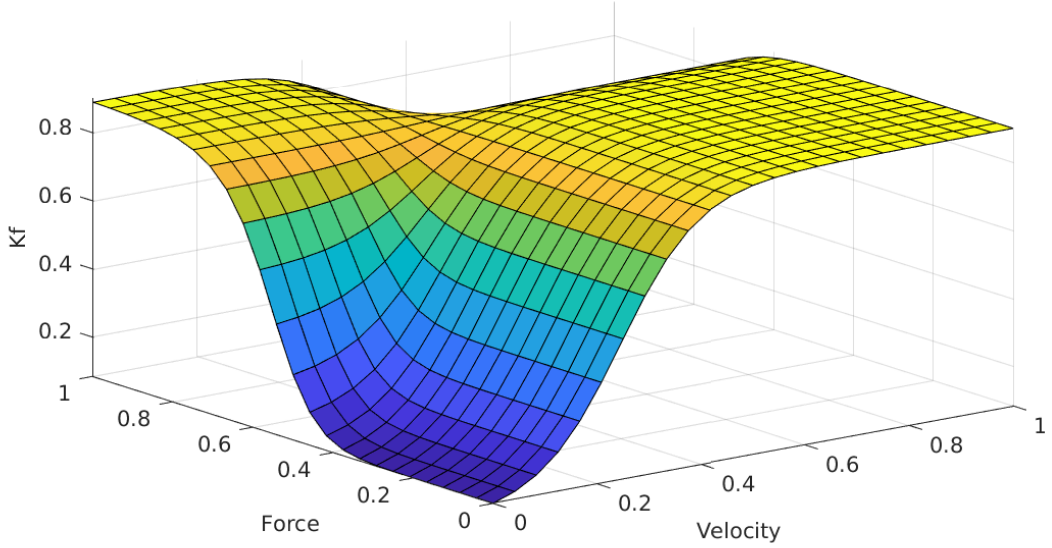


Figure 31: FIS surface for $\nu_i = \nu_b = 1.1$ for membership function $\mu_{\hat{F}_{ext},3}$ (*'high'*).

To identify the best fuzzy surface with force contribution we will look for ν_i values of the MF $\mu_{\hat{F}_{ext},3}$ that minimize ITAEP and ITAF in the simulation with McRuer's model for different values of $d_i(s)$.

Figure 32 displays the two surfaces s_1 and s_2. The first one represents the normalized values $\hat{ITAEP} = \frac{ITAEP}{\max(ITAEP)}$ (top surface) while the second one represents the normalize values $\hat{ITAF} = \frac{ITAF}{\max(ITAF)}$ (low surface) obtained.

It is observable how the ITAF is greater compared with ITAEP. Considering both surfaces $s_1(\nu, d)$ and $s_2(\nu, d)$, and $(\nu, d) \in [\nu_0, \nu_{N-1}] \times [d_0, d_{N-1}]$, it is possible to define two new surfaces:

$$\begin{aligned} f_1(\nu, d) &= \max\{s_1(\nu, d), s_2(\nu, d)\} \\ f_2(\nu, d) &= s_1(\nu, d) + s_2(\nu, d) \end{aligned} \tag{64}$$

Both surfaces are represented in Figure 33 (blue surface for $f_1(\nu, d)$ and yellow surface for $f_2(\nu, d)$). The minima points of $f_1(\nu, d)$ and $f_2(\nu, d)$ correspond to minimum value of ITAEP and ITAF for different disturbances d. MFs parameters values corresponding to this points can be considered good choice for the FIS.

In order to have only one value for the MF, we can proceed as follows: considering function $f_1(\nu, d)$, it is possible to describe a function $g_1(\nu, k)$ to define all possible absolute minima at different fixed values of d such as:

$$g_1(\nu, k) = \begin{cases} z = \min f_1(\nu, d) \\ d = k \; \forall k \in [d_0, d_{N_d - 1}] \end{cases} \tag{65}$$

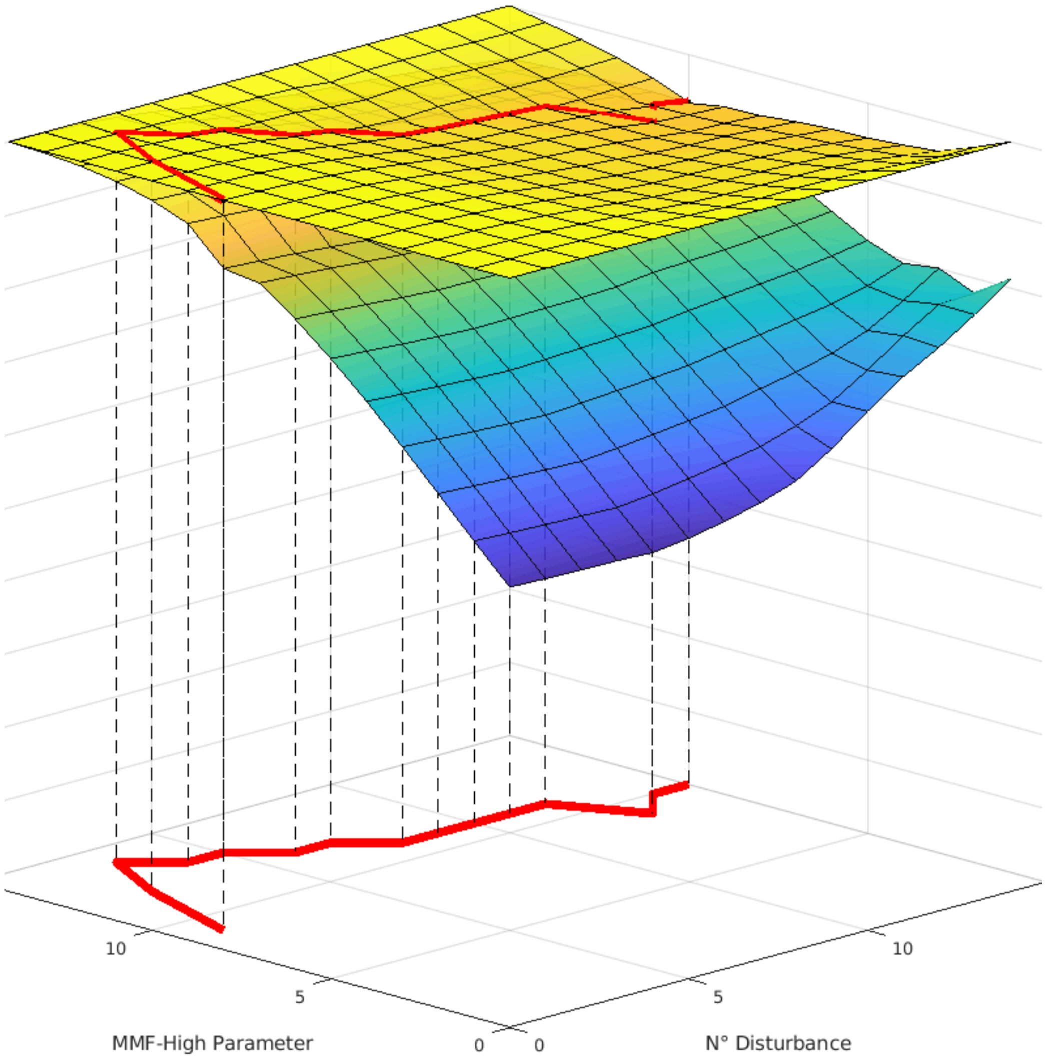


Figure 32: Surfaces $s_1(\nu, d)$ and $s_2(\nu, d)$ as a function of ν and external disturbances. Both functions represent ITÂEP and ITÂF (normalized values of ITAEP and ITAF). In red: $g_1(\nu, k)$, a function which defines the absolute minima of $f_1(\nu, d)$ for different values of d fixed.

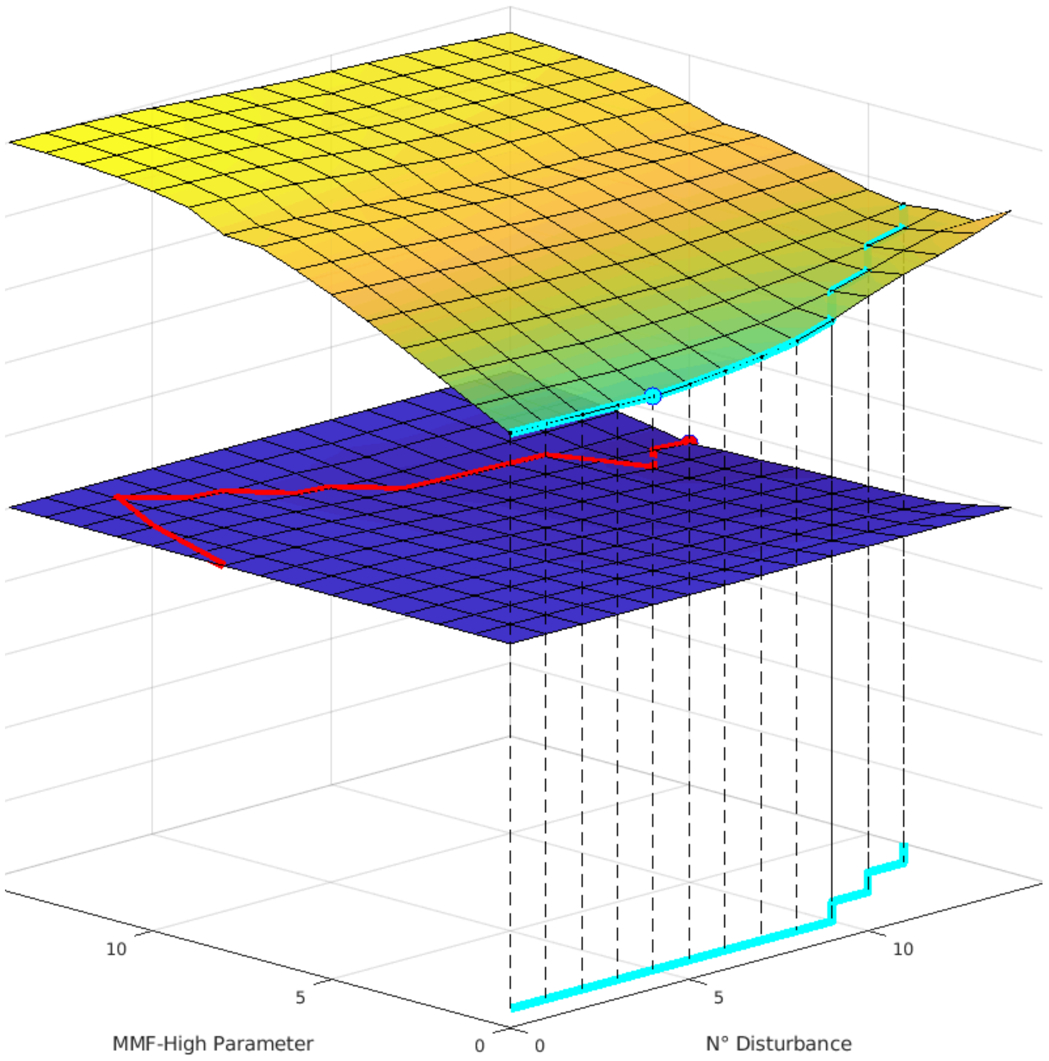


Figure 33: Functions $f_1(\nu, d)$ (blue surface) and $f_2(\nu, d)$ (yellow surface)representing, respectively, the max value between ITÂEP and ITÂF and the sum of both FOM. In red: $g_1(\nu, k)$, the function which defines the absolute minima of $f_1(\nu, d)$ at d fixed. In cyan: $g_2(\nu, k)$, the function which defines the absolute minima of $f_2(\nu, d)$ at d fixed.

The function $g_1(\nu, d)$ is represented in Figure 33 as the red curve.

The absolute minimum (ν_0, k_0) along g_1 is obtained such that:

$$g_1(\nu_0, k_0) = \min g_1(\nu, k) \tag{66}$$

All these considerations are valid also for $f_2(\nu, d)$. In Figure 33, $g_2(\nu, k)$ with a cyan line. $g_1(\nu_0, k_0)$ and $g_2(\nu_0, k_0)$ are represented with colored dots. From Figure 32 it is possible to evince how ITAEP is almost constant across the surface, on the contrary ITAF present a significant change. A single value of $\nu = \nu_1 = 0.6$ is able to provide the minimum values of ITAF until a disturbance almost equal to $d_{10} = 9[N]$, as represented by the cyan curve in Figure 33, and can be a good choice for the MF parameters. In ideal case, if the disturbance could be precisely recognized, only the MF with parameter $\nu_1 = 0.6$ could be used. In practice, since the disturbance is variable, a mean value of the function $g_1(\nu, d)$ (cyan curve in Figure 33) is preferable. In conclusion, we chose the MF which parameter $\nu = 1,32$ which corresponds to the mean ν value of $g_1(\nu, d)$.

2.2.6.1 *Simulation Results*

In terms of simulation results, considering the MF calculated as discussed in previous section, we compared the obtained FIS controller with the FIS only-velocity scaling and the linear scaling. As illustrated in Figure 34 The combined force-velocity FIS further reduces the values of ITAF. In particular, considering a simulation test with $N_d = 3$ in Equation (50), the force-velocity FIS controller ITAF reducing is equal to 39.7%, 38.4%, 20.3% compared with the only-velocity scaling and equal to 60.2%, 53.7%, 28.1% compared with the linear case.

2.3 COMPARISON OF IMPEDANCE CONTROL-BASED ALGORITHMS

The proposed elasto-plastic control along with fuzzy control based on a human intention model are two approaches to impedance control. Although such, they have differences that make them better suited for certain purposes and less so for others depending on which of the two models is to be used. In the elasto-plastic model when the robot encounters forces below a certain level, it behaves like a spring, becoming more resistant as the movement increases. However, if the forces exceed this threshold, the robot's behavior becomes more rigid, eliminating any bounce-back effect and allowing for permanent changes in shape. Tests on these models reveal hat different operators may prefer different impedance tuning to accomplish the same task, and this can be achieved by implementing a gain scheduling. Fuzzy control, on the other hand, does not exhibit a sharp separation between elastic and plastic behavior; rather, FIS produces

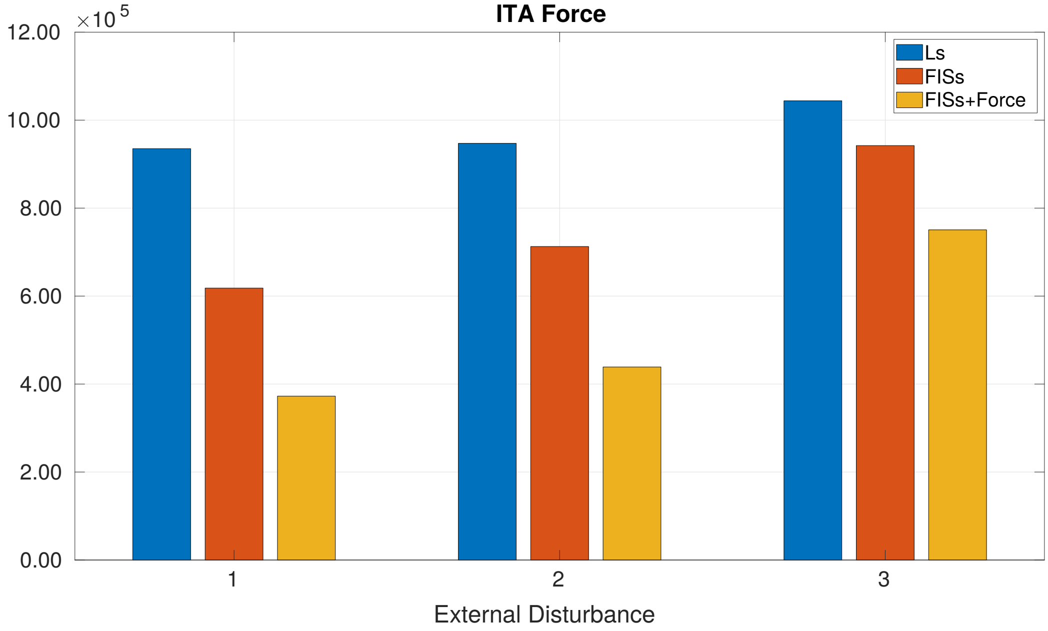


Figure 34: ITAF comparison. FIS control applied with $N_d = 3$ disturbances considered. The blue bars represent the FIS linear scaling, the red bars represent the FIS velocity scaling, the yellow bars represent the force-velocity FIS scaling. The force required by the controller (human) is strongly reduced in the force-velocity FIS case.

impedance modulation based on force and velocity at which the pHRI task is performed. Simulated tests show that a combination of these two and a properly optimized FIS lead to reduced force in the operator and improved position tracking. We could then divide the two algorithms according to certain qualities of the pHRI. Advantages and disadvantages of each algorithm are cataloged in Table 4 and reported here:

- **motion**: includes the motion nature of object velocity and acceleration felt by the human when a given control strategy is applied on the robot.
  If the fuzzy-based control is not accurately tuned, too rapid changes in force and velocity along the fuzzy surface can result in a fast impedance change, giving the operator the feeling of having little control over the robot. In contrast, the elasto.-plastic control has only two modes with fixed impedance parameters (net of gain scheduling) therefore a rapid change in dynamics is not present;

- **smoothness**: whether the robot movement is smooth or note during the pHRI.
  Fuzzy-based control presents smoother motion since there is no sudden transition between two modes. In contrast, the bump between the two state transitions in the elasto-plastic can cause discomfort in the operator;

Table 4: Advantages and disadvantages of impedance control-based algorithms.

	Motion	Smoothness	Performance	Effort	Versatility
Fuzzy-based Control	-	+	-	+	-
Elasto-plastic Control	+	-	+	+	+

- **performance**: minimizing the robot position error and forces involved can be considered as an indicator of the good performance of a pHRI-focused control.
  Elasto-plastic control, having two dedicated modes of operation, has better performance in both. Taking a precise manual guidance operation as an example, the elastic phase remains with the same values even if the operator applies forces within a certain range. Conversely, if an unexpected force is applied to the fuzzy system this can cause a rapid deviation from the path.

- **effort**: the physical exertion required by the operator when performing a pHRI task with a given control strategy;
  It is possible to say that in this case both methods produce advantages. The operator's effort in the elasto-plastic mdoel is reduced due to the absence of a repulsive component. In the fuzzy system, however, it is low, since it is possible to vary the impedance rapidly.

- **versatility**: how much the same control strategy can be applied to several different tasks (i.e. robot manual guidance, pick$place, etc.
  The two operating modes make it a very versatile elastic-plastic model. For example: we can use the plastic mode while moving the robot effortlessly in manual guidance, grab an object with the robot. Carry out assembly operations facilitated by the elastic part.)

3 3D SCANNER SYSTEM FOR CASTING INSPECTIONS

According to Industry 4.0 paradigms, contemporary industrial systems should incorporate a variety of methods that combine traditional industrial automation with digital advancements. Furthermore, there is a requirement for economical digital automation strategies that can be easily integrated into existing manufacturing operations. Quality inspection with non-contact sensors and relative data collection, can be framed as an innovative solution that is seen as a central aspect industry 4.0. In this context this thesis proposes two computer vision based applications, that may be useful in manufacturing and beyond.

The market offers 3D scanners stations for advanced measuring and metrology that are employed to examine components either by sample or at the conclusion of the manufacturing process. However, production line inspection systems are less common: implementing real-time monitoring on the production line would enable the detection of any flaws on the component during the early stages of production. Here, an inspection system that can be the basis for an inline inspection system has been realized. The system consists of an optical setup using a 3D structured light laser scanner (SLS) and a software (SW) with an intuitive interface (GUI) that can be easily used by the user.

The system designed is developed for applications in die casting, and it is able to detect defects on mechanical components molded in this way. However, it can be easily extended to other applications as well.

The 3D scanner acquire a 3D model of the structural casting component to be inspected and compare it with a 3D reference model. The reference can be the ideal CAD model of the analyzed part, or another part (previously acquired) and considered good. In this work we consider the CAD model of the analyzed part as the reference. The 3D model acquired is spatial point cloud, which is basically a discrete set of x, y, z data points in space.

This makes it possible to evaluate differences in flatness and defects between the acquired part and the reference model, but also to have information about the x, y, z coordinates of the part.

The system allows to perform the following operations:

- acquisition of a 3D "sample" scan of a casting component placed on a bench;
- compare the scan with a "reference" model: the CAD or a previously acquired model;

- generate a heatmap to highlight dimensional differences (defined as ΔE) between the two models and related defects.

Each step of the procedures is automatic, excluding a calibration step, which is currently manual but performed only once at system startup, the pipeline of the procedure is visible in Figure 35.

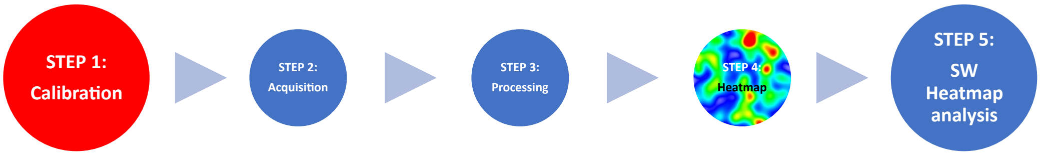


Figure 35: Inspection system pipeline.

There are two acquisition steps:

- projection of a laser pattern onto the surface of the object;
- the deformation induced by the surface of the object is acquired by a camera and exploited to calculate three-dimensional coordinates.

3.1 STRUCTURED LIGHT SCANNER 3D LASER SCANNER

SLS is a 3D reconstruction technology base on triangulation. Instead of using a laser to emit dots or lines, like in laser triangulation, this scanner employs a projector that projects a fringe pattern onto the surface being measured. The laser pattern, projected into the camera image plane, has no ambiguities, so this scanner are able to generate a high quality 3D point cloud from each image acquired at good scanning speeds.

As with all optical methods, highly reflective or transparent surfaces make the acquisition process problematic for structured light. Reflection could create back light directly in the camera optics bringing them into saturation, or a reflection on the object itself would compromise the regularity of the pattern. In either case, the acquisition is deteriorated. Transparent or semitransparent surfaces cause even greater difficulties, as the projected shapes do not produce recognizable profiles on the surface of the object.

In this work we used a Photoneo PhoXi 3D Scanner size M(Figure 36) [79]. The datasheet of the scanner is shown in Table 5, and the working space in Figure 37.

Figure 36: Photoneo PhoXi 3D Scanner (size M).

Table 5: Photoneo PhoXi 3D Scanner (size M) datasheet.

Parameter	Value
Scanning range	458 - 1118 mm
Optimal scanning distance (sweet spot)	650 mm
Scanning area (at sweet spot)	590 x 404 mm
Point to point distance	0.286 mm
Calibration accuracy (1σ)	0.100 mm
Temporal noise (1σ)	0.100 mm
Scanning time	250 - 2500 ms
Dimensions	77 x 68 x 416 mm
Baseline	350 mm
Weight	950 g

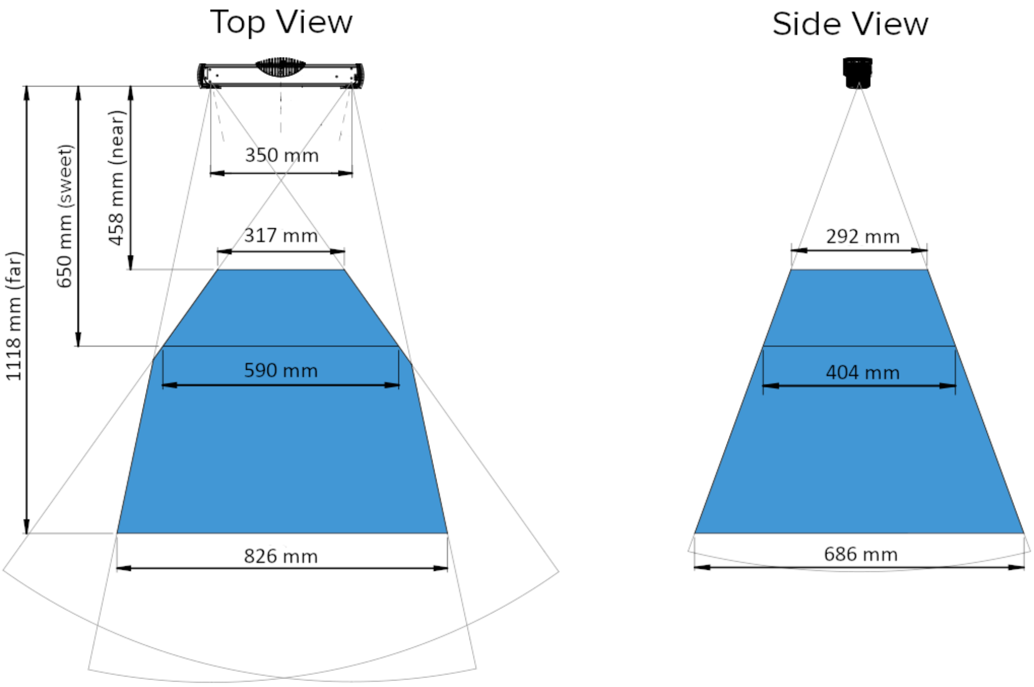


Figure 37: Photoneo PhoXi 3D Scanner (size M) working space.

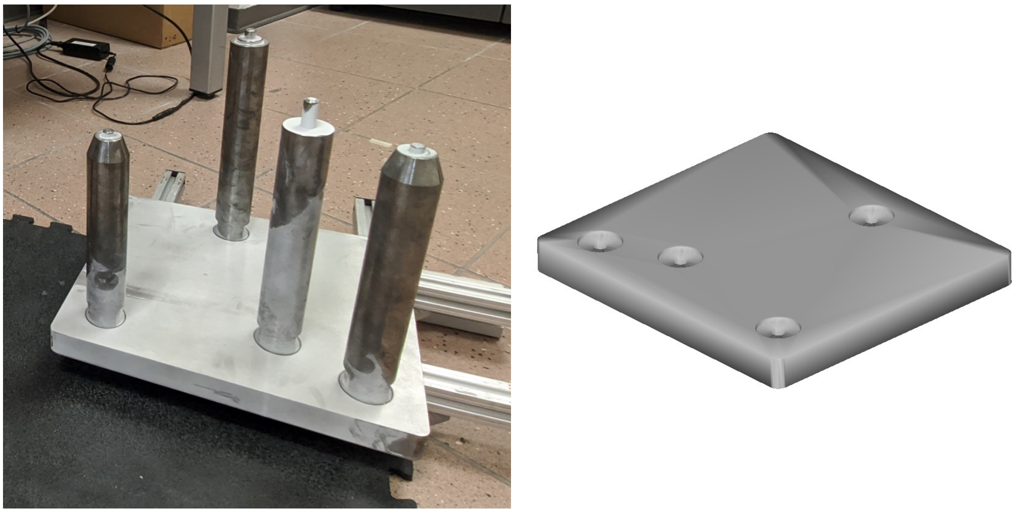


Figure 38: Baseplate made of $Fe360$ and $39NiCrMo3$ (left). Size: $300x330[mm]$. The relative CAD model used as reference (right).

Among all the specifications it is worthy of mention that the scanner has:

- calibration accuracy: $0.1[mm]$ @ working distance of $650[mm]$;
- minimum point-to-point distance: $0.286[mm]$.

A small accuracy is very important in die-casting applications where accurate and repeatable measurement is critical for proper quality control. In addition, the minimum point-to-point distance ensures a high-density point cloud.

An baseplate made of $Fe360$ and $39NiCrMo3$ (Figure 38) with dimensions $300x330[mm]$ was 3D acquire with the SLS and used for system performance analyses. Such a baseplate is made with tolerances $< 0.1[mm]$ and is therefore considered a good reference for evaluating the measurement accuracy of the proposed system. It is also equipped with a series of columns on which a die-cast part can eventually be placed. From the point of view of a production line, the base can be tailored to the part to be inspected, the system is calibrated on this base, and the parts to be inspected will be placed on it and correctly measured once alignment is performed, since part and base will form a single body. The movement of the part can be performed through a robotic pick&place, or a small automated station.

3.2 ALIGNMENT PROCEDURE

Inspection without physical contact involves comparing two point clouds (the sample and the reference) in different reference system. Hence, prior to the comparison, the objects being analyzed must be aligned with one another. We consider the point cloud reference already placed in the reference with 3 orthogonal planes between them whose intersection coincides with the origin. Since our model is a parallelepiped and it is a CAD model, this pre-positioning is very straightforward. In general this alignment procedure could be called calibration, since it is the first procedure that must be performed before making a new measurement with the software. In the quality control of mechanical components, the design specifications for feature location and tolerance are determined with respect to a fixed, reproducible reference system. A Cartesian reference system is commonly used in inspection and can be defined using the 3-2-1 alignment rule [80]. The importance and effectiveness of this method has been emphasized in [81]. Basically, according to this rule, a plane, a line, and origin point are used to establish the directions of the sample reference system that will be aligned on the reference one. Figure 39 illustrates the overall procedure: the user selects 6 keypoints (3 for plane, 2 for line, 1 for point) and the software perform the alignment.

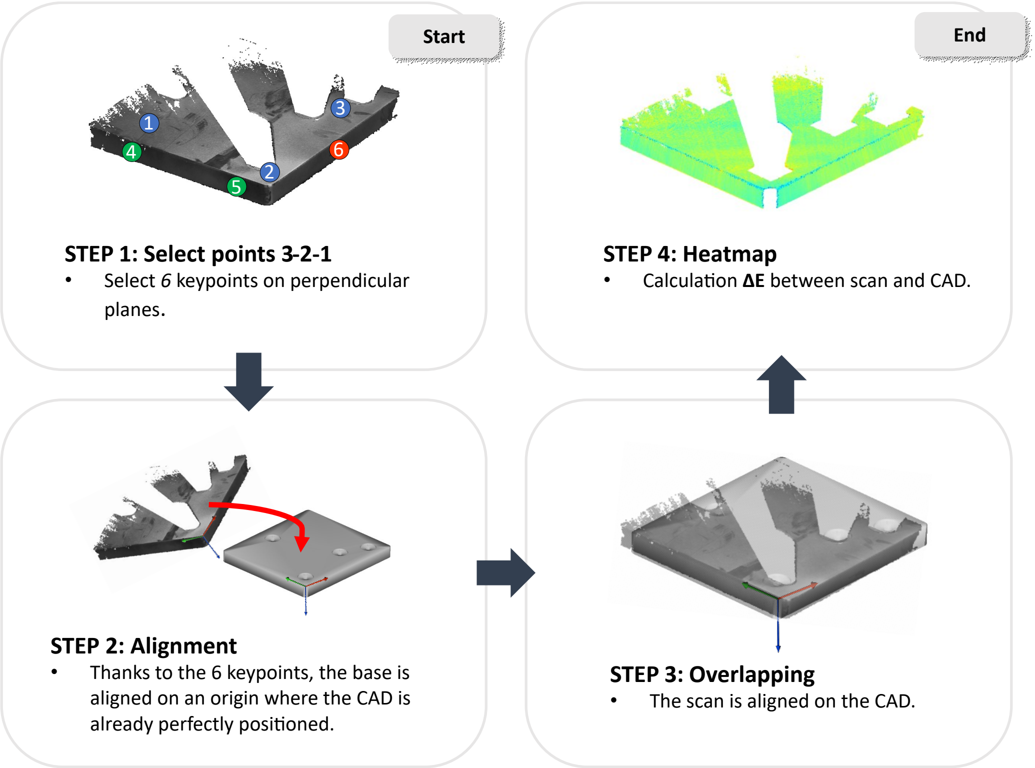

Start
1
2
3
4
5
6
STEP 1: Select points 3-2-1
• Select 6 keypoints on perpendicular planes.
End
STEP 4: Heatmap
• Calculation ΔE between scan and CAD.
STEP 2: Alignment
• Thanks to the 6 keypoints, the base is aligned on an origin where the CAD is already perfectly positioned.
STEP 3: Overlapping
• The scan is aligned on the CAD.

Figure 39: Alignment procedure: the user selects 6 keypoints (3 for plane, 2 for line, 1 for point) and the software perform the alignment.

With these keypoints we can find 3 planes that define origin and axis on the new reference system. Each element is identified under the following conditions:

1. plane1: identified by at least 3 points. The equation coefficients can be obtained by a least square method.

2. plane2: passes through at least 2 points (line) and it must be parallel to the plane1. The equation coefficients can be obtained by a least square method under the constraint of parallelism.

3. plane3: passes through at least 1 point and it must be parallel to the plane1 and plane2. The equation coefficients can be obtained by a least square method under the constraint of parallelism.

Once the planes have been identified, it is straightforward to identify the points which outlining the axes of the new reference system.

Considering Figure 40, we can define some vectors describing the points $\mathbf{P_x}$, $\mathbf{P_y}$, $\mathbf{P_z}$ in the Euclidean space:

$$\mathbf{P_y} = \mathbf{P_{1,prj}} - \mathbf{P_0'} \qquad \mathbf{P_x} = \mathbf{P_{2,prj}} - \mathbf{P_0'} \qquad \mathbf{P_z} = \mathbf{P_x} \times \mathbf{P_y} \tag{67}$$

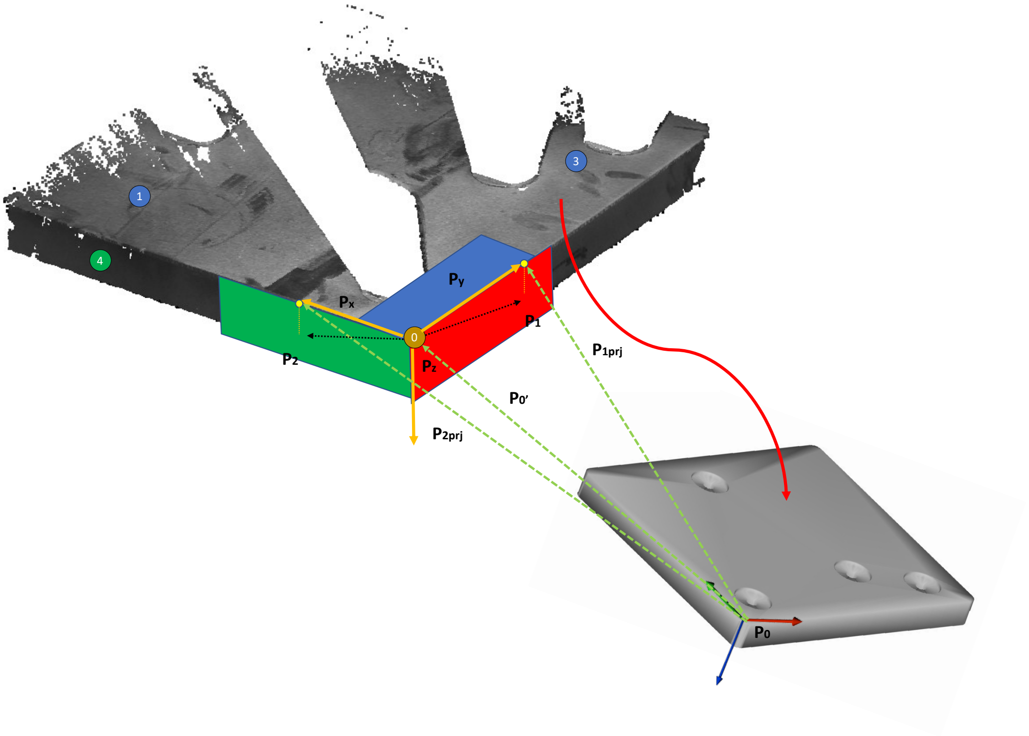


Figure 40: Calibration. Change of reference system of the acquired point cloud.

And, considering $\xi = \mathrm{atan2}(P_{x1}, P_{x0})$, $\theta = \mathrm{atan2}(P_{x2}, P_{x0})$, $\phi = \mathrm{atan2}(P_{z2}, P_{z1})$ it is possible to define the rotation matrix to change reference system:

$$\mathbf{R_z} = \begin{bmatrix} \cos\xi & \sin\xi & 0 \\ -\sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{R_y} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}, \mathbf{R_x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}$$

$$(68)$$

And the overall rotation matrix becomes:

$$\mathbf{R_0} = \mathbf{R_x R_y R_z} \tag{69}$$

Since a translation to the origin is necessary, the final rototranslation matrix is computed as:

$$\mathbf{R} = \begin{bmatrix} R_{0,00} & R_{0,01} & R_{0,02} & -P'_{0,x} \\ R_{0,10} & R_{0,11} & R_{0,12} & -P'_{0,y} \\ R_{0,20} & R_{0,21} & R_{0,22} & -P'_{0,z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{70}$$

Eventually, by transforming any general point $\mathbf{P}'$ to a new point $\mathbf{P}$ such as $\mathbf{P} = \mathbf{RP}'$, we can considered the point cloud aligned on the reference.

This procedure is based on a CAD reference with 3 perfectly perpendicular planes. In general, one can pre-position a sample on the refence taking care that in both models the selected planes are the same. However, to achieve a more robust alignment, the developed software has a best-fit function that allows the sample to self-align once it is rotated around the reference part.

3.3 HEATMAP ANALYSIS

After the alignment is done correctly, the 3D model acquire is assumed to be perfectly positioned over the CAD model. In this way, the Euclidean distance ΔE between the points of the two meshes can be calculated. There are two general methods (Figure 41) for calculating the cloud-cloud distance, as specified in [3]:

- cloud-to-cloud: calculate the Euclidean distance between a point on the sample cloud and the nearest neighbour in the reference cloud;
- cloud-to-mesh: calculate the Euclidean distance between one point and the perpendicular of the nearest polygonal face of the reference mesh.

As specified in [3], the "cloud-to-cloud" presented higher deviations than "cloud-to-mesh." This result can be explained by a geometrical approach. The perpendicular distance between one point and one mesh is always the shortest. Moreover, in a comparison of perfectly planar surfaces, in case the density of points is small, a cloud-cloud comparison would lead to very large distances. In contrast, the presence of a mesh would allow distance to be calculated even with respect to "empty" areas of points. For these reasons, cloud-to-mesh (Figure 42) distance was used in our system.

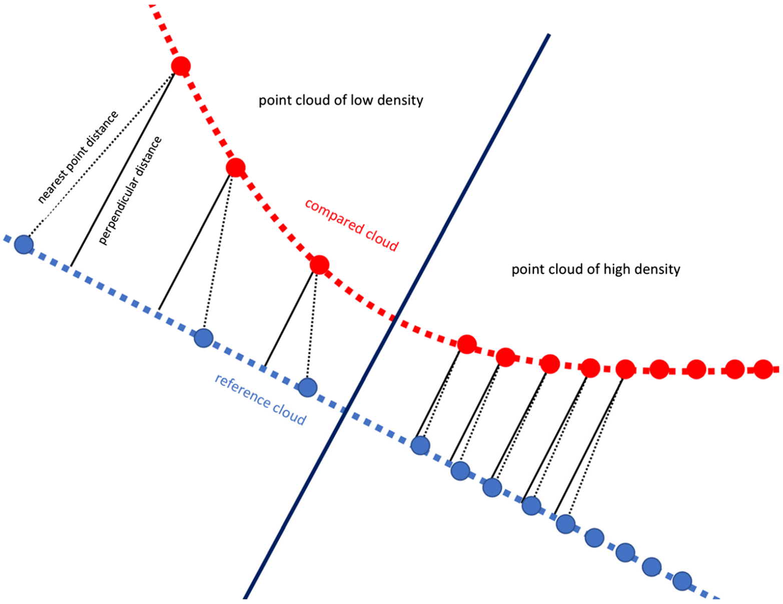


Figure 41: Method to calculate point cloud distance: cloud-to-cloud and cloud-to-mesh [3].

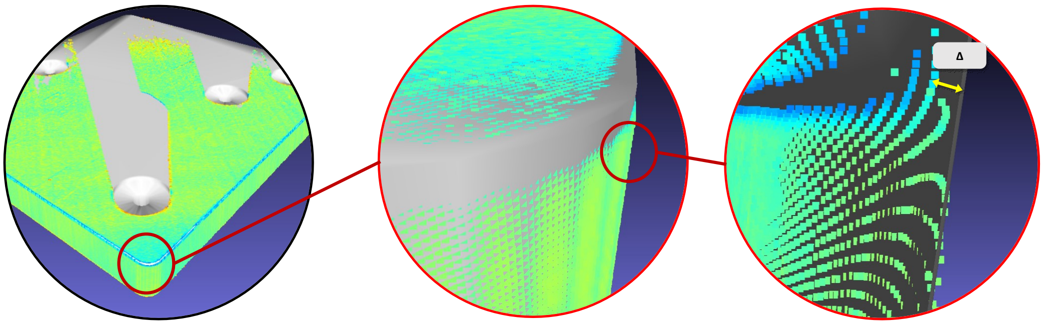


Figure 42: Cloud-to-mesh distance applied at our 3D model.

3.3.1 *Evaluation of accuracy and repeatability*

An optical setup has been designed to evaluate accuracy and repeatability, as shown in Figure 43. The scanner is able to capture three perpendicular faces of the base at the same time. The datasheet recommended measurement distance is $650[\mathrm{mm}]$ from the scanner (sweat spot). Between $550 - 750[\mathrm{mm}]$ we have the best scanning accuracy. To evaluate performance even on a small scan volume, we acquire two scans:

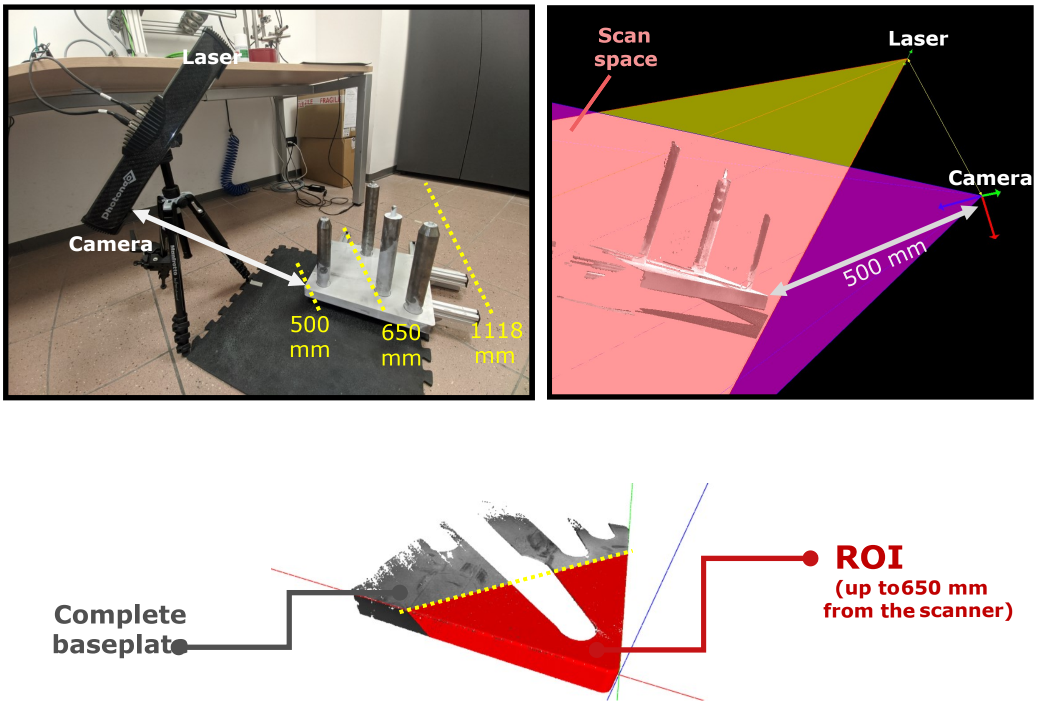


Figure 43: Setup to evaluate accuracy and repeatability.

- a full scan of the base;
- a region of interest (ROI) of the base between 500 and 650 mm away from the scanner (area with best accuracy).

The scanner was able to acquire a point cloud of $300,000$ points. After the alignment procedure, we performed the cloud-to-mesh distance. The resulting mesh is visible in Figure 44. We can consider ΔE as an indicator on the maximum scan error. In Table 6, are visible the values of ΔE calculated on the totality of scan points.

To calculate ΔE_{max} and ΔE_{σ}, possible outliers (Figure 45) were removed, such as:

- sharp edges or corners (not present in CAD as it is not beveled);
- small clusters (accumulated matting paint, tiny debris etc. They are $< 5\%$ of the points in the cloud).

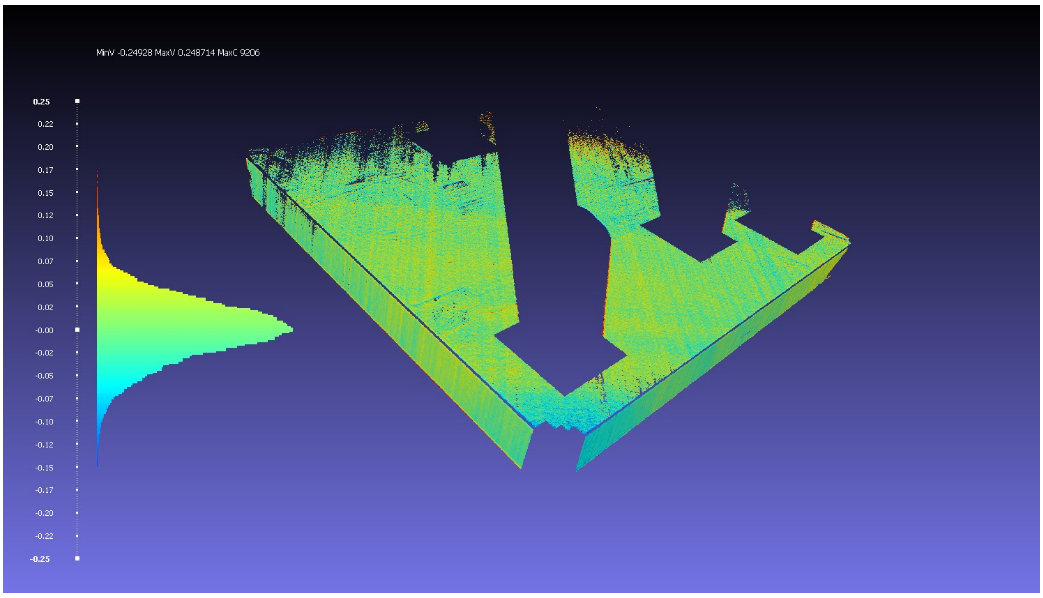


Figure 44: Resulting heatmap after alignment on the CAD model and cloud-to-mesh distance calculation. The distance of all the points are visible on the histogram on the left. The majority of the points have a distance ΔE in the range $\pm 0.1[mm]$ and the overall standard deviation $\Delta E_{\sigma} = 0.037[mm]$.

Table 6: Maximum and standard deviation of ΔE among all the points of the acuired point cloud.

	$\Delta E_{max}[mm]$	$\Delta E_{\sigma}[mm]$
Normal sample	0.085	0.037
ROI sample	0.067	0.031

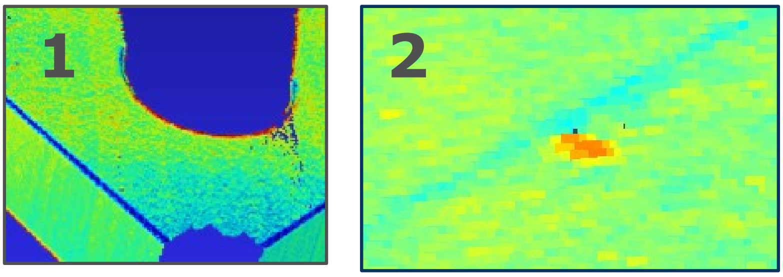


Figure 45: Possible scanner basement outliers, such as sharp corners and small clusters of debris.

Table 7: Repeatability results. The standard deviation is very low, with a maximum value equal to 21.98[µm] on the Z coordinate.

Points	X[mm]	Y[mm]	Z[mm]	Norm [mm]
Pnt1	-9.538	61.870	658.651	661.619
Pnt2	-9.538	61.869	658.638	661.606
Pnt3	-9.538	61.868	658.632	661.600
Pnt4	-9.537	61.867	658.619	661.587
Pnt5	-9.538	61.870	658.646	661.614
Pnt6	-9.538	61.870	658.650	661.618
Pnt7	-9.537	61.866	658.604	661.572
Pnt8	-9.537	61.865	658.591	661.559
Mean[mm]	-9.538	61.868	658.629	661.597
σ[mm]	0.318	2.064	21.985	22.084

A test to assess the repeatability of the scanner was also carried out. Eight point clouds were acquired in succession (Figure 46). For each new one, a small portion was taken. Of that portion, for each point, the distance between that point and the nearest point from the other acquisitions was analyzed.

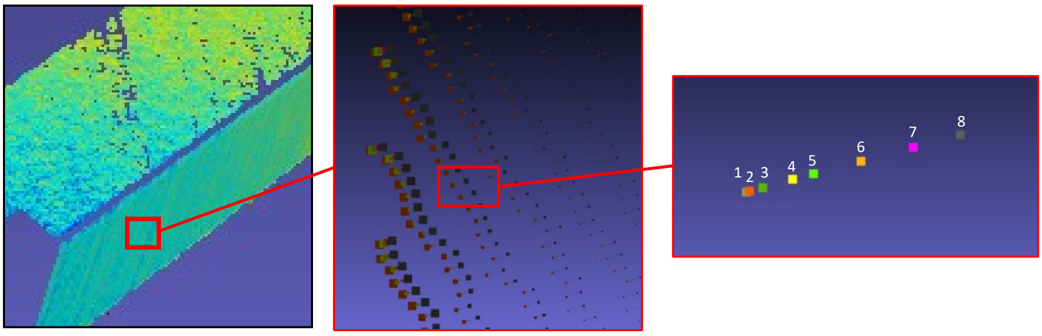


Figure 46: Eight point clouds were acquired in succession to verify repeatability.

The standard deviation has a maximum value equal to 21.98[µm] on the Z coordinate. This is consistent with what is stated in the datasheet as Temporal Noise (Z-noise): the Gaussian noise on the Z-coordinate of a single point acquired (the Z being the axis exiting the scanner camera). The table 6 shows a $\Delta E < 0.1$[mm], a result we can consider acceptable for rapid measurement applications in die-casting and demonstrating that system alignment and processing are performed correctly with low uncertainties.

3.4 SOFTWARE AND GUI

The software is designed to have a quick, direct and immediate approach to the user. Accordingly, the graphical user interface (GUI) is designed to have an intuitive interface that is easily used by a nonexpert user directly in the work area where quality control is performed. The software consists of two macro modules:

- **Engine**: written in C++ language. It consists of the DLLs in which the algorithms that form the core of the software are executed: drivers to comunicate with the Phoxy 3D Scanner, alignment algorithms, Euclidean distance map algorithms, I/O management, data storage and management;
- **GUI**: written in C# language. constitutes all the user interface part. It handles the loading of 3D models, visualization, and state logic that commands what and when something is displayed on the screen.

the SW has an autorun mode and a manual mode. In the first case, the part to be analyzed is loaded and analyzed automatically; it is assumed that the logic to manage this workflow is through external PLC and triggers. The second mode allows the operator to set a new recipe and perform the analysis manually. We will try to describe the latter better as it allows us to illustrate all the components of the SW and how a measurement and analysis cycle takes place. The main panels of the GUI are as follows:

3.4.1 *Acquisition panel*

It is the initial page with which the user is approached (Figures 47,49,50,51). It is divided into two side panels.

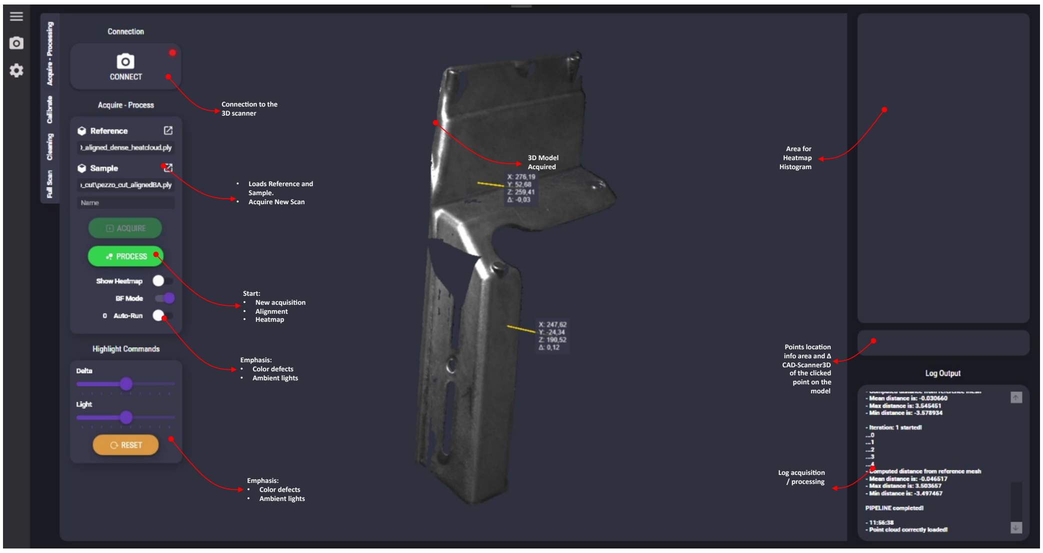
Left panel: allows you to perform 3D model management operations and start the processing algorithm:

- connect to the 3D scanner and display its current status via LED (ON/OFF);
- upload a 3D model in .PLY format of both the reference and the sample model;
- acquire a new sample model to compare with the reference model, via "Acquire" button;
- process the model, i.e. apply the distance map algorithm for the quality control check via the "Processing" button, which produces a heatmap where the part is highlighted;

- An enhancement panel that allows you to emphasize the colors of the distance map once the heatmap is produced; this helps the user give qualitative information about the part they are analyzing;
- increase the illumination of the 3D model for better visualization, through 3D post-processing techniques.

Right panel: allows to view different output information:

- A central viewer to visualize the loaded/acquired model;
- A viewer to display the histogram of the heatmap that was calculated by processing;
- A viewer to display the X,Y,Z location of a point cloud selected by the user. Delta information will also be available for that point;
- A logger to have a recap of the last activities performed.



Reference: ›_aligned_dense_heatcloud.ply

Sample: ›_cut\pezzo_cut_alignedBA.ply

Connection to the 3D scanner

- Loads Reference and Sample.
- Acquire New Scan

Start:
- New acquisition
- Alignment
- Heatmap

Emphasis:
- Color defects
- Ambient lights

Emphasis:
- Color defects
- Ambient lights

3D Model Acquired

Area for Heatmap Histogram

Points location info area and Δ CAD-Scanner3D of the clicked point on the model

Log acquisition / processing

```
- Computed distance from reference mesh
- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934

- Iteration: 1 started!
...0
...1
...2
...3
...4
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467

PIPELINE completed!

- 11:56:38
- Point cloud correctly loaded!
```

Figure 47: SW acquisition panel. The user can manage all phases related to model measurement and analysis, including: loading, processing, enhance defects.

3.4.2 *Calibration panel*

In this panel (Figure 48), the user can perform a calibration, it consists of the following panel:

Left panel: allows you to perform several operation on a point cloud:

- load a the model on which the calibration will be performed;

- calibrate by clicking on the model by selecting the 6 points (3 for a plane, 2 for a line, 1 point), visible in green in Figure 48, on which the 3-2-1 method will be used. it is possible to choose 3 modes of point selection, to have different robustnesses:
  - single point: exclusively selects the point selected in the point cloud;
  - local neighbors: once a point is selected, it takes a sample of n neighboring points and extracts their average x,y,x coordinates. This helps to have some robustness in case of noise;
  - global planes: automatically selects all the portion of points matches the given plane and extracts the equation of the plane that best fits these points. It allows for the highest level of robustness since all points in a plane are used instead of local ones.
- insert a custom calibration matrix: allows you to enter a calibration matrix with custom value.

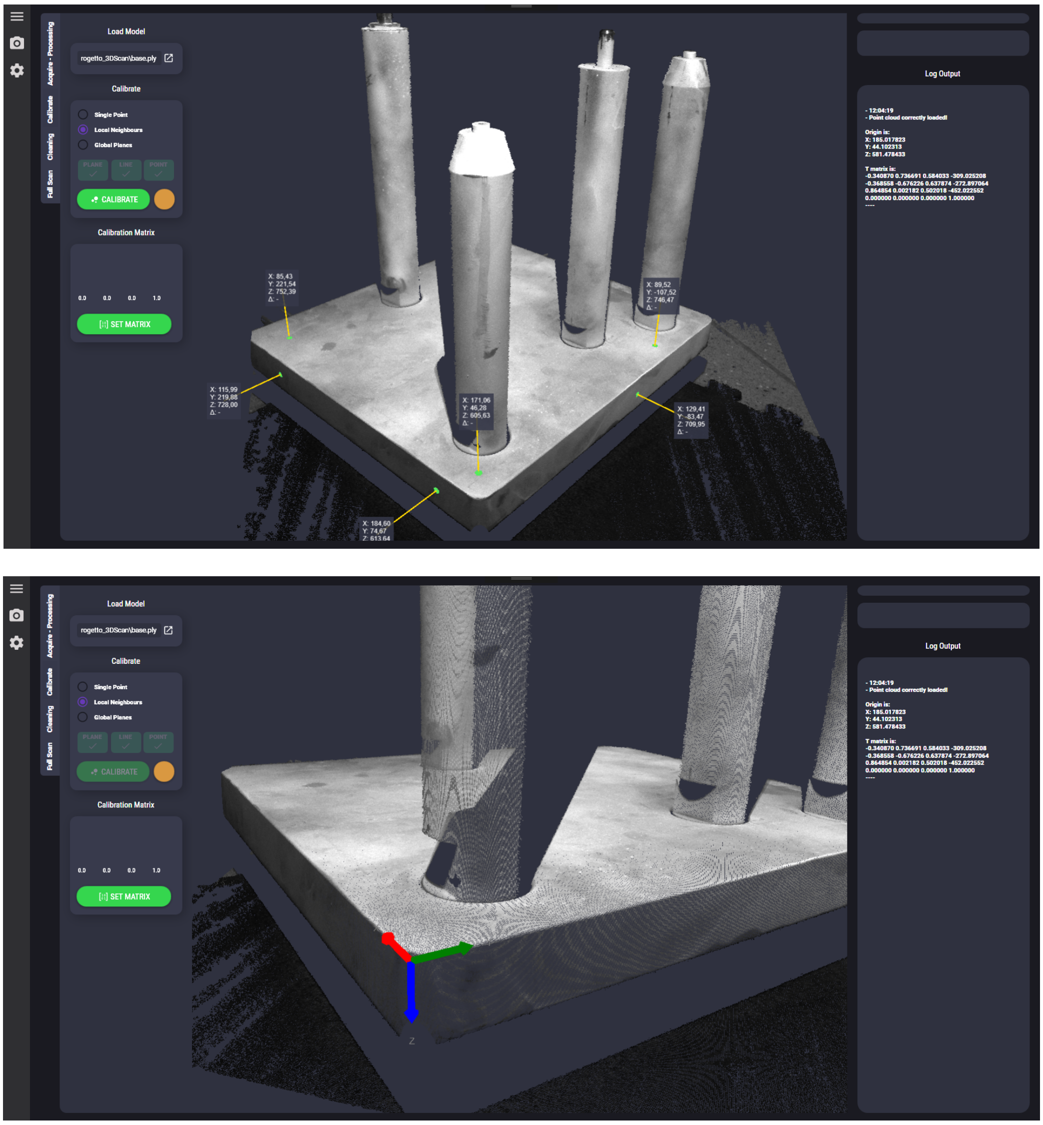
3.4.2.1 *User Interface: make a new recipe*

If the user wants to create and execute a new recipe, meaning the acquisition of a new model, the user must perform the following steps in succession. First the calibration must be done, in the "Calibration panel":

1. *"Load Model button"*: loads the model on which you want to perform the calibration;
2. *"Point selection"*: the user selects all keypoints to apply rule 3-2-1;
3. *"Calibrate button"*: Apply the 3-2-1 method to calculate the rototranslation matrix (calibration matrix) between the model origin the origin 0. The model will automatically rototranslate by positioning itself on the new origin as in Figure 48 (bottom).

Now the system is calibrated, all newly acquired models will be automatically rototranslated relative to the calibration matrix. If the component to be analyzed was of a different type, a new rototranslation has to be produced by calculating a new calibration matrix. It is then possible to proceed with acquiring a new model to be measured and related processing:

1. *"Camera button"*: automatically detect the IP address of the scanner and connect to it.
2. *"Load reference button"*: loads a reference template.
3. *"Load/Acquire sample button"*: load/acquire a sample template



```
- 12:04:19
- Point cloud correctly loaded!

Origin is:
X: 185.017823
Y: 44.102313
Z: 581.478433

T matrix is:
-0.340870 0.736691 0.584033 -309.025208
-0.368558 -0.676226 0.637874 -272.897064
0.864854 0.002182 0.502018 -452.022552
0.000000 0.000000 0.000000 1.000000
----
```

- 12:04:19
- Point cloud correctly loaded!

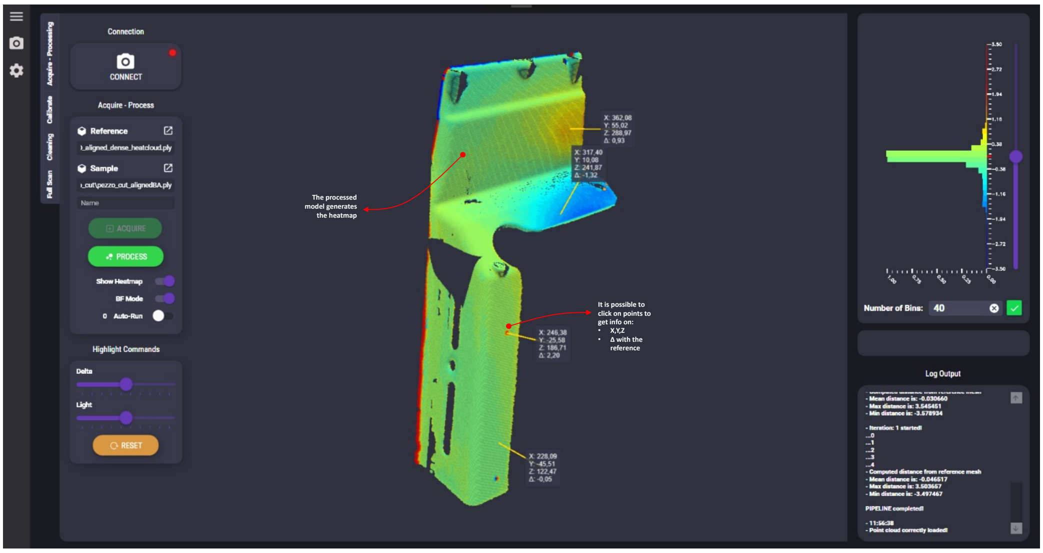
Origin is:
X: 185.017823
Y: 44.102313
Z: 581.478433

T matrix is:
-0.340870 0.736691 0.584033 -309.025208
-0.368558 -0.676226 0.637874 -272.897064
0.864854 0.002182 0.502018 -452.022552
0.000000 0.000000 0.000000 1.000000

Figure 48: SW calibration panel. The user can calibrate the component with the 3-2-1 rule.

4. *"Process button"*: run the distance map algorithm and produces a heatmap result. The result is as in Figure

The end result is as visible in the Figure 49: the user can have qualitative information about the quality of the inspected of the part. Parts that deviate too much from the reference will be highlighted in blue or red (if the deviation is negative or positive). it is also possible to have information about individual points, by clicking on the model, related to x, y, z position of the point and value of the corresponding ΔE.



Reference:)_aligned_dense_heatcloud.ply

Sample:)_cut\pezzo_cut_alignedBA.ply

Name

Show Heatmap ●

BF Mode ●

0 Auto-Run ○

The processed model generates the heatmap

It is possible to click on points to get info on:
- X,Y,Z
- Δ with the reference

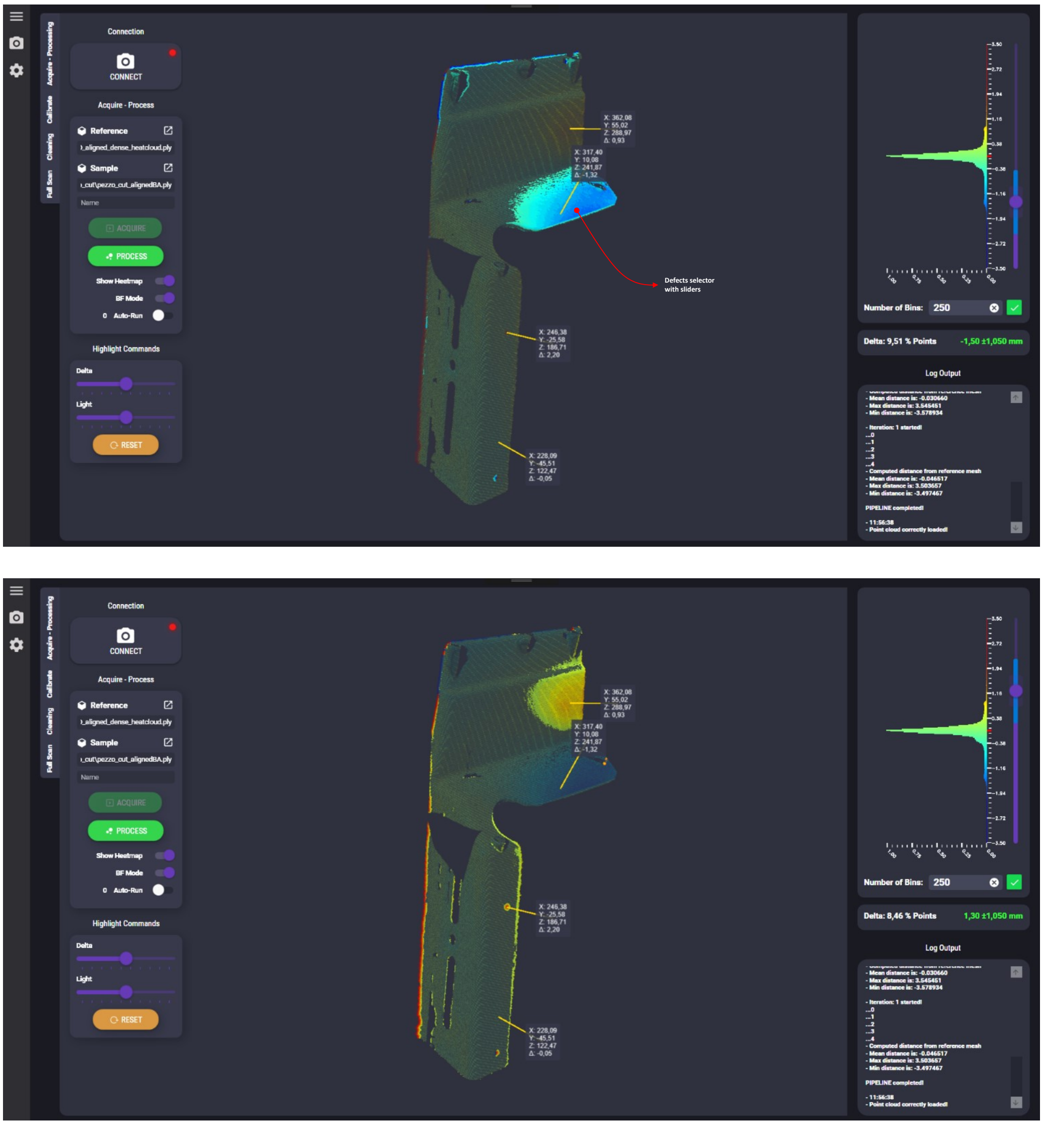
Number of Bins: 40

```
- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934

- Iteration: 1 started!
...0
...1
...2
...3
...4
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467

PIPELINE completed!

- 11:56:38
- Point cloud correctly loaded!
```

Figure 49: SW acquisition panel. Processing result. It is possible to see the heatmap and by click to have information about the point such as x, y, z coordinates and ΔE.

Figure 50 shows the ΔE ROI (region of interest) mode for easier visualization of defects. it is in fact possible to act on the histogram in the right panel to highlight only the portion of the ΔE object of interest.

At the same time, it is possible to enhance the visualization of defects through color space change or changes in light spot and shadows by acting on the left panel "Highlight Commands" as shown in Figure 51.



Reference:)_aligned_dense_heatcloud.ply

Sample: ›_cut\pezzo_cut_alignedBA.ply

Name

Show Heatmap ●

BF Mode ●

0 Auto-Run ○

```
X: 362,08
Y: 55,02
Z: 288,97
Δ: 0,93

X: 317,40
Y: 10,08
Z: 241,87
Δ: -1,32

X: 246,38
Y: -25,58
Z: 186,71
Δ: 2,20

X: 228,09
Y: -45,51
Z: 122,47
Δ: -0,05
```

Number of Bins: 250

Delta: 9,51 % Points -1,50 ±1,050 mm

```
- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934

- Iteration: 1 started!
...0
...1
...2
...3
...4
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467

PIPELINE completed!

- 11:56:38
- Point cloud correctly loaded!
```

Reference:)_aligned_dense_heatcloud.ply

Sample: ›_cut\pezzo_cut_alignedBA.ply

Name

Show Heatmap ●

BF Mode ●

0 Auto-Run ○

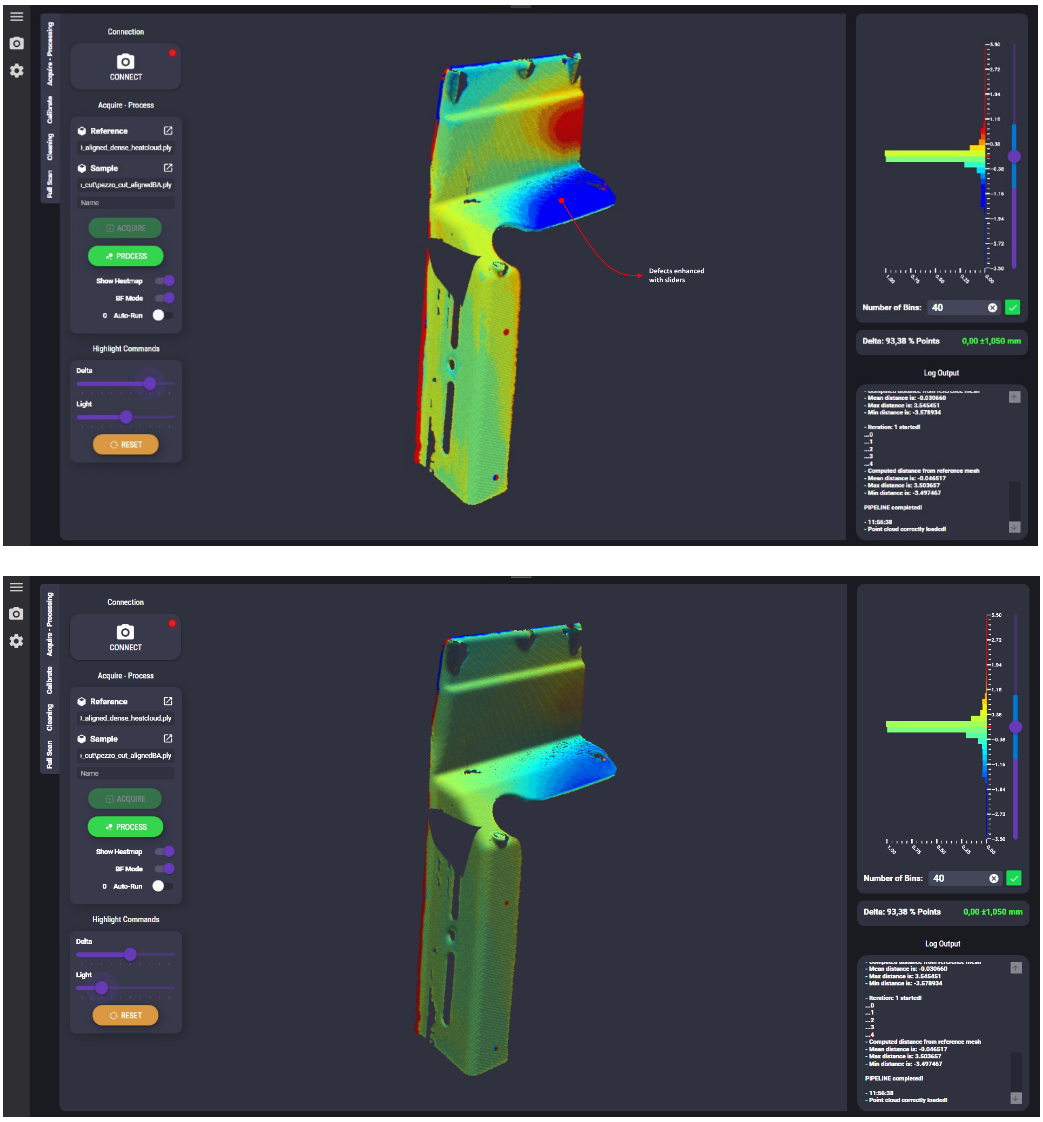
```
X: 362,08
Y: 55,02
Z: 288,97
Δ: 0,93

X: 317,40
Y: 10,08
Z: 241,87
Δ: -1,32

X: 246,38
Y: -25,58
Z: 186,71
Δ: 2,20

X: 228,09
Y: -45,51
Z: 122,47
Δ: -0,05
```

Number of Bins: 250

Delta: 8,46 % Points 1,30 ±1,050 mm

```
- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934

- Iteration: 1 started!
...0
...1
...2
...3
...4
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467

PIPELINE completed!

- 11:56:38
- Point cloud correctly loaded!
```

Figure 50: SW acquisition panel. Defects selection mode.



Reference:)_aligned_dense_heatcloud.ply

Sample: ›_cut\pezzo_cut_alignedBA.ply

Name

Show Heatmap ●

BF Mode ●

0 Auto-Run ○

Delta

Light

Number of Bins: 40

Delta: 93,38 % Points 0,00 ±1,050 mm

```
- Computed distance from reference mesh
- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934

- Iteration: 1 started!
...0
...1
...2
...3
...4
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467

PIPELINE completed!

- 11:56:38
- Point cloud correctly loaded!
```

Reference:)_aligned_dense_heatcloud.ply

Sample: ›_cut\pezzo_cut_alignedBA.ply

Number of Bins: 40

Delta: 93,38 % Points 0,00 ±1,050 mm

- Mean distance is: -0.030660
- Max distance is: 3.545451
- Min distance is: -3.578934
- Iteration: 1 started!
- Computed distance from reference mesh
- Mean distance is: -0.046517
- Max distance is: 3.503657
- Min distance is: -3.497467
PIPELINE completed!
- 11:56:38
- Point cloud correctly loaded!

Figure 51: SW acquisition panel. Defects enhancement mode.

4 EVM SYSTEM FOR FREQUENCY DETECTION

The Eulerian Video Magnification (EVM) [4] method can reveal amplitude movements that may be difficult or impossible for humans to perceive. As discussed in [4] the basic approach is to consider the time series of color values at any spatial location (pixel) and amplify variation in a given temporal frequency band of interest. Other attempts have been made to unveil imperceptible motions in videos, analyzing and amplifying subtle motions and visualize deformations that would otherwise be invisible [55, 82]. These approaches follow a *Lagrangian* perspective, in reference to fluid dynamics wherethe trajectory of particles is tracked over time. In contrast, EVM use an *Eulerian* approach, where for each voxel of fluid, the evolution of properties such as density, velocity, pressure is analyzed. In this case, the voxel becomes a pixel; the EVM focuses on each pixel and "photographs" the flow of intensity variation in that pixel.

Thanks to the *Eulerian* approach what flows into the pixel over time is not just a color variation of a static scene. In fact, if we consider a scene with an object subject to micro mechanical vibrations, the information content of a pixel framing the object will change color not because of actual color variation of the object itself, but because the object is moving in the scene. In summary, EVM makes it possible to identify color changes and micromovements in the scene. To clarify, inn Eulerian approach the motion is not explicitly estimated, instead it is rather exaggerate by amplifying temporal color changes at fixed positions.

The EVM use a combination of spatial and temporal filtering to highlight delicate temporal variations in a video. The video sequence is separated into various spatial frequency levels with a complete Laplacian or Gaussian pyramid. As they can have varying signal-to-noise ratios, they may be amplified differently. After spatial sampling, a temporal processing is carried out on each spatial level. Afterwards, the extracted bandpass signal is increased by a user-specified factor α. Lastly, the amplified signal is added to the original image and the spatial pyramid is collapsed to produce the final output.

This approach combines spatial and temporal processing to highlight subtle temporal changes in a video. The video sequence is first broken down into different spatial frequency bands, which may be magnified differently based on their signal-to-noise ratios. In general, a full Laplacian pyramid is computed. Next, temporal processing is applied to each spatial band, with the processing being uniform for all bands and pixels within each band.

4.0.1 *Spatial filtering*

To accomplish this method, temporal filtering must be applied to lower spatial frequencies (spatial pooling). In this way the input signal can rise above the noise from the camera sensor and quantization conversion. Additionally, when the objective of this spatial processing is to enhance the accuracy of the signal over time by combining multiple pixels, a low-resolution filter on the video frames can improve computational performance. In our case spatial filtering is accomplished by means of a Laplacian or Gaussian pyramid [83]. Taking into account a Gaussian pyramid (Figure 52), images are scaled down using a Gaussian blur and reduced in size. Each pixel representing a neighborhood average corresponds to a nearby pixel on a lower level of the pyramid. A Laplacian pyramid is very similar to a Gaussian pyramid but saves the difference between the blurred images at each level. The highest resolution image is reconstructed by utilizing the difference images from the upper levels, with the exception of the lowest level."

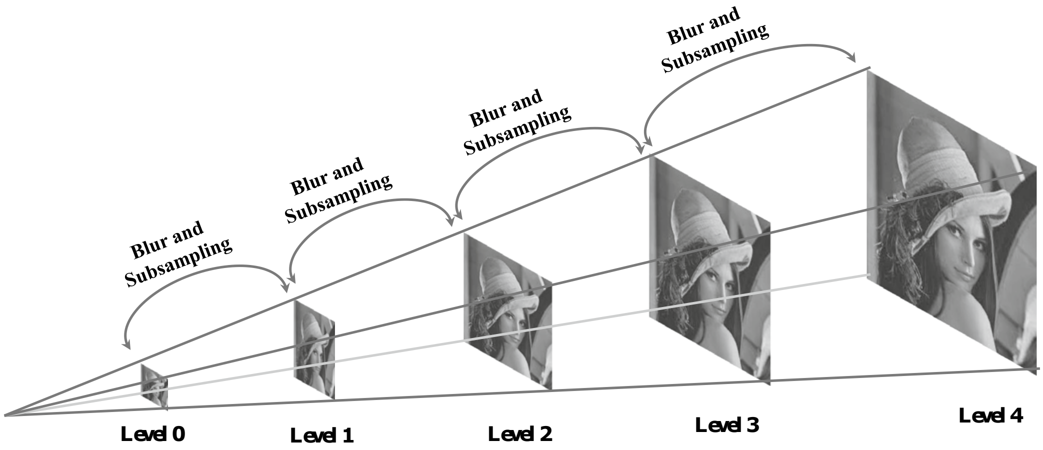
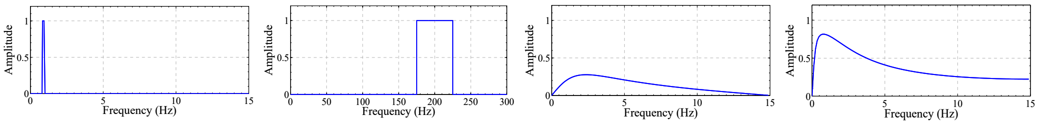


Figure 52: Representation of a Gaussian pyramid with 5 levels.

4.0.2 *Temporal filtering*

A temporal bandpass filter is used to isolate and amplify the desired signal or motion. The choice of filter is determined by the specific application. For example, when magnifying motion, a filter with a broad passband is preferred, while for amplifying blood flow color, a filter with a narrower passband is more effective in reducing noise. Figure 53 illustrates the frequency responses of some of the temporal filters. Typically, low-order IIR filters [84, 85] can be useful for both color amplification and motion magnification and are practical for real-time implementation. In this work, two first-order lowpass IIR filters with cutoff frequencies ωl and ωh to construct an IIR bandpass filter were used.



(a) Ideal 0.8-1 Hz (face) (b) Ideal 175-225 Hz (guitar) (c) Butterworth 3.6-6.2 Hz (subway) (d) Second-order IIR (pulse detection)

Figure 53: Examples of temporal filters used form EVM [4].

4.0.3 *Signal extraction*

Now, to isolate and amplify the desired signal several steps are necessary:

- choose a temporal bandpass filter;
- chose the value of the amplification factor α;
- choose a temporal bandpass filter;
- Decide on a spatial frequency cutoff (determined by spatial wavelength, λ_c) at which a reduced version of α is applied;
- Select the method of reduction for α either completely eliminating it for all $\lambda < \lambda_c$, or gradually decreasing it to zero.

Once you have obtained the amplified video signal, it is possible to visualize the output video signal in which variations in color or vibration will be evident. The realized software allows to select a region of interest (ROI) in the source video, calculates its color average over the three RGB channels, reduce them to a mean monochrome channel and extract a a 1-dimensional data by the spatial average of this channel. At this point we obtain a new 1-dimensional signal $s(t)$ on which the FFT will be calculated to extract the principal frequency components.

4.1 TEST AND PERFORMANCE

Overall, there are 2 possible macro areas of focus for this algorithm:

Healthcare

- heart rate detection;
- respiratory rate detection.

Industrial and manufacturing

- detection of mechanical vibrations, invisible to the eye.

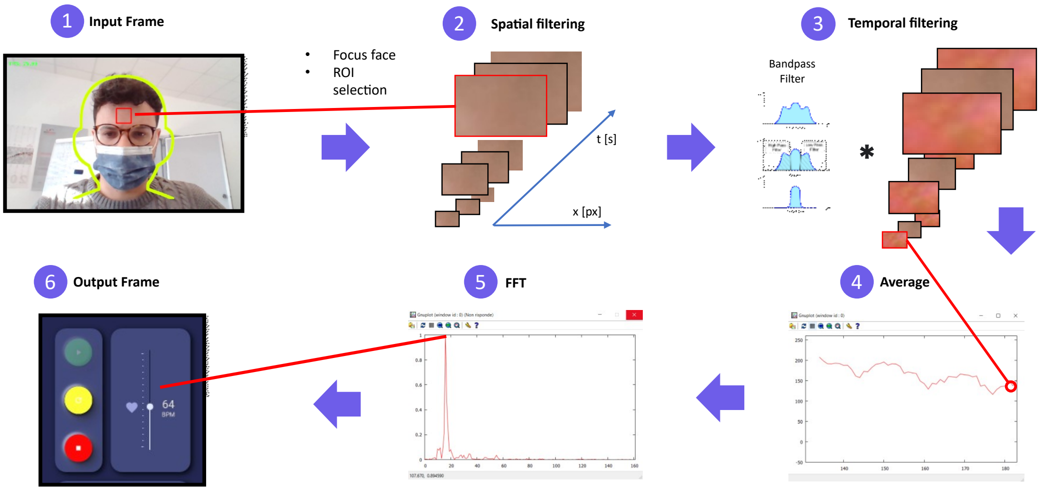


Figure 54: EVM pipeline for heartbeat detection. 1) ROI selection, 2) Spatial filtering, 3) Temporal filtering, 4) color average over the three RGB, 5) FFT of the average signal 6) Detected heartbeat in the GUI.

Since this work was developed in a scenario conditioned by COVID-19, most of the focus was given to the healthcare part, particularly heartbeat detection. So as to develop a useful application for self-monitoring and tracking user's health status. However, the system can be used int the same way to monitor micro variations in mechanical components and analyze their vibrations in order to collect data for quality control or predictive maintenance perspective. Moreover, from preliminary results, the heartbeat detection is the most sensitive to noise and external disturbances. This helped deepen this issue, including designing a strategy to optimize robustness.

4.1.1 *Heartbeat monitoring*

To extract the heartbeat detected values, the pipeline in Figure 54 was followed. A circular buffer with multiple collected frames is used to apply the algorithm, at each new frame the first frame of the buffer is erased and a new one in pushed in. We chose to use the red channel in the RGB color space which is more informative regarding the hemoglobin content of the skin. The time buffer of 2D frame is collpased to 1D pixel buffer using the spatial mean among each pixel of red channel, in order to have a 1D data. On this data it has been calculated the spatial FFT. To extract the heartbeat (defined as the beat per minute BPM), from the result of the spatial FFT, the following formula is required:

$$BPM = 60 \frac{n_{px} N_{FPS}}{n_{px,max}} \tag{71}$$

Where n_{px} is i-th pixel of the 1D image FFT, $n_{px,max}$ is the lenght in pixel of the 1D image FFT, and N_{FPS} is the number of frame processed each second (FPS). TO calculate the FFT we used a Hamming mobile window with overlap 99% and sime size of the circular buffer of frames. The speed and performance of the algorithm make it unnecessary to use minor overlap.

The window size L_w will go to determine the spectral spatial resolution of the FFT df_{BPM}, and thus the minimum BPM we can identify, which results:

$$df_{BPM} = 60\frac{N_{FPS}}{L_w} \tag{72}$$

For heartbeat application, the following set of EVM parameters has been used.

Algorithm parameters:

- Gaussian Pyramid levels: 5;
- amplification parameter α : 5;
- frequencies range (ω_l, ω_h): $55 - 130[BPM](0.91 - 2.16[Hz])$;
- buffer size: 450 frames;
- Hamming window size: 450 frames.

Using a L_w=450 frames we obtain a df_{BPM}=4. (uncertainty on detection of ± 4 BPM). This turns out to be a first limitation of the algorithm. Considering a video acquisition with a standard camera at FPS=30 we obtain that a window of 450 frames corresponds to about 15 [s] of acquisition time in which the subject must necessarily remain still. Therefore, the trade-off between L_w and df_{BPM} has to be evaluated. From the first tests performed, the parameters specified above appear to be a fair trade-off.

4.1.1.1 *Considerations about robustness*

To increase the robustness of the algorithm in case of external light variation (due to slight movement of the face or light source), an approach proposed is as follows: assuming the modulations of external light intensity is the same for all RGB channels, a ratio of two normalized color channels would not be affected by external light movement. The pulse frequency of blood would still be available in this ratio provided that the pulsation due to changes in blood volume is different in individual color channels. Given hemoglobin's high absorption of light in wavelengths near green and low in red, the ratio of the normalized green and red channels would emit a robust pulse such that:

$$S_{i,j} = \frac{G_{i,j}}{R_{i,j}}\frac{\mu(R_{i,j})}{\mu(G_{i,j})} - 1 \tag{73}$$

Where i, j are the i-th and j-th pixel in a generic image. The signal thus obtained will be the one used as an alternative to the monochrome channel to increase the robustness of the algorithm. A preliminary test was performed with the $S_{i,j}$ signal and the monochrome channel. In the second case an unwanted motion creates a rapid increase (or decrease) in brightness, altering the frequency signal and showing unwanted components in the FFT. The $S_{i,j}$ signal presents a similar situation but with more attenuated unwanted amplitudes, demonstrating the benefit in using the latter.

4.1.1.2 *Performance test*

A total of 9 tests were performed, in different locations and illuminations on 1 subject. A standard PC's internal camera was used to record test videos. This camera records videos at 720@30FPS, the use of higher performance cameras can further improve the results obtained. The best illumination conditions were then repeated on 2 additional subjects. Video signal acquisition was performed in real-time via the PC's internal camera. Exposure and white balance are automatically adjusted by the camera. The following lights (Figure 55) were used at different times to illuminate the scene: 1 Neon light, and 3 LED lights. The lights are directed as follows:

- two directions of the light source: front and side to the face;
- two types of illumination: direct (LED2) and diffuse (Neon, LED1, LED3).

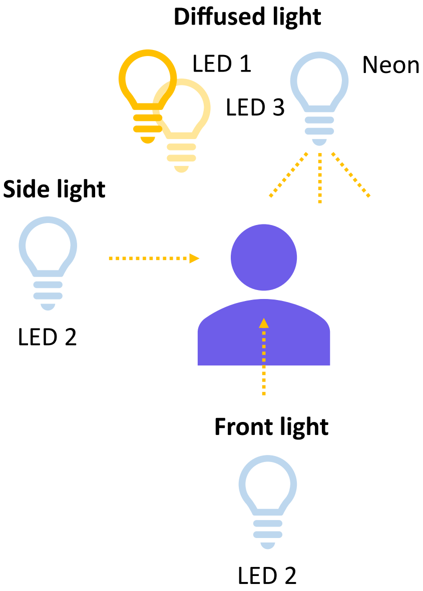


Figure 55: Different light sources to test the performance of EVM while detecting heartbeat.

Each test involves an acquisition period of 450 frames (1 buffer window) + 150 additional frames. Acquiring on average at 30FPS this is equivalent

Table 8: Results of the tests with different light conditions.

Test	Ligh Type	FPS	BPM alg	BPM smart	conf.	dBPM	$\text{dBPM}_{\%}$
1	Frontal Neon	30	87	90	79%	3	4%
2	Side Neon	28	87	91	77%	4	5%
3	Frontal LED1	15	73	77	82%	4	5%
4	Frontal LED2	28	77	79	65%	2	3%
5	Side LED2	28	71	77	75%	6	8%
6	LED2 Low Light	19	96	76	62%	20	27%
7	Frontal LED3	28	72	73	73%	1	1%
8	Frontal Natural Light	29	65	64	73%	1	1%
9	Side Natural Light	29	66	64	79%	1	1%

to a measurement time of 20[s]. For each test performed, BPM values detected by the algorithm were compared with those detected by a wrist smartband having fitness/health tracking capabilities. The smartband detection time is 30[s], in which the wrist must necessarily be stationary. The results from the smartband are used as benchmarks for the algorithm's measurements. Given a video capture time of 15[s] (450 frames), the algorithm is potentially faster in providing an estimate of BPM. For each test we calculate:

- FPS : FPS of the camera;
- BPM_{EVM}: heartbeat detected by EVM;
- BPM_{smart}: heartbeat detected by the smartband;
- *conf*: a custom confidence value of the algorithm, obtained by averaging the maximum peaks detected by the FFT. This helps in identifying whether the maximum peak is an isolated peak and helps estimate a highly disturbed FFT from a cleaner one;
- $dBPM = |BPM_{EVM} - BPM_{smart}|$: difference between heartbeat detected by the EVM algorithm and the smartband;
- $dBPM_{\%} = \frac{dBPM}{\omega_h - \omega_l}$: relative value of $dBPM$. Where ω_h and ω_h are the max/min frequency range of the temporal filter measured in BPM.

Results of the tests are visible in Table 8. Overall, the results did not differ much from those detected by the smartband, with maximum error detected in tests 5 and 6. Both tests were performed with poor or partial lighting level. In test 6 the average FPS is lower comapred to other tests. This is largely caused by limited ambient lighting: the PC camera has auto exposure, in dark environments this causes an automatic increase in exposure time, slowing down the algorithm and generating low and variable FPS. In addition, the ISO sensitivity is automatically increased, making the image noisier. These factors generate a poorly clean FFT and uncertain measures. Tests 8 and 9 have the best performance, they have been carried out in natural light. Tests 1-2-3 report a good result with higher average confidence levels. It is common to all tests performed having good performance the low level of noise present in the image, a condition of greater impact in the performance of the algorithm. The natural illumination tests were tested on 2 other subjects, obtaining the same level of performance.

4.1.2 *Mechanical vibration monitoring*

The pipeline in Figure 54 was used again to extract some valuable information related to mechanical vibrations. These are low amplitude, high frequency mechanical vibrations. In this case we was limited by the capture rate of the camera. With a standard acquisition@30FPS it is possible to detect small movements at frequencies < 15[Hz], without incurring aliasing phenomena. A slow-motion acquiring software was therefore used to record video@240FPS, In this case the ROI is taken on a small portion of the image where a possible vibration can occur. A high FPS video of a car engine (Renault Clio) was then recorded, as shown in Figure 56.

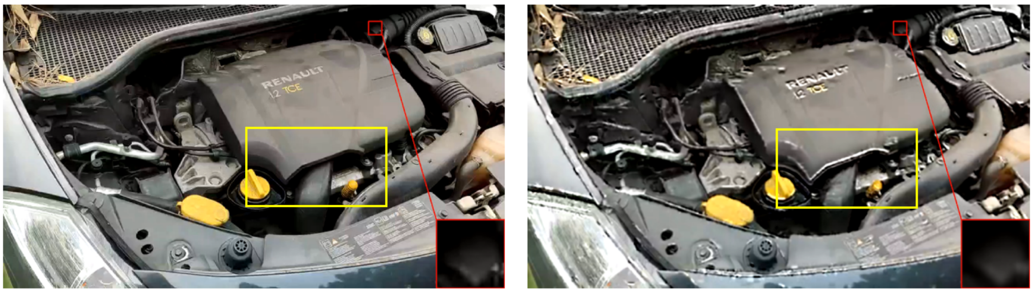


Figure 56: EVM applied to an engine. In yellow, a ROI of the video in which signal amplification can be observed (on the right), such as a change in pixel color: in fact, at some sharp edges of the engine, they become brighter. This chance in color can be detected to extract frequency information.

The same pipeline in Figure 54 was followed. As for hearthbeat detection, a circular buffer with multiple collected frames is used to apply the algorithm, at each new frame the first frame of the buffer is erased and a new one in pushed in. We chose to use the RGB color space and calculate the FFT on the average of all the channel to obtain a monochrome channel. The aim was to identify the engine rotation frequency detected by a component in contact with the engine itself. For the algorithm we used the following parameters.

Algorithm parameters:

- Laplacian Pyramid levels: 5;
- amplification parameter α : 20;
- frequencies range (ω_l, ω_h): $15 - 25[\text{Hz}]$;
- buffer size: 450 frames;
- Hamming window size: 450 frames.

From this measurement we were able to extract a video signal of the average trend of the pixels of a specific ROI, by applying an FFT on the time signal of this ROI, in correspondence with the maximum peak it is possible to trace the value of rotation per minute of the motor (Figure 57).

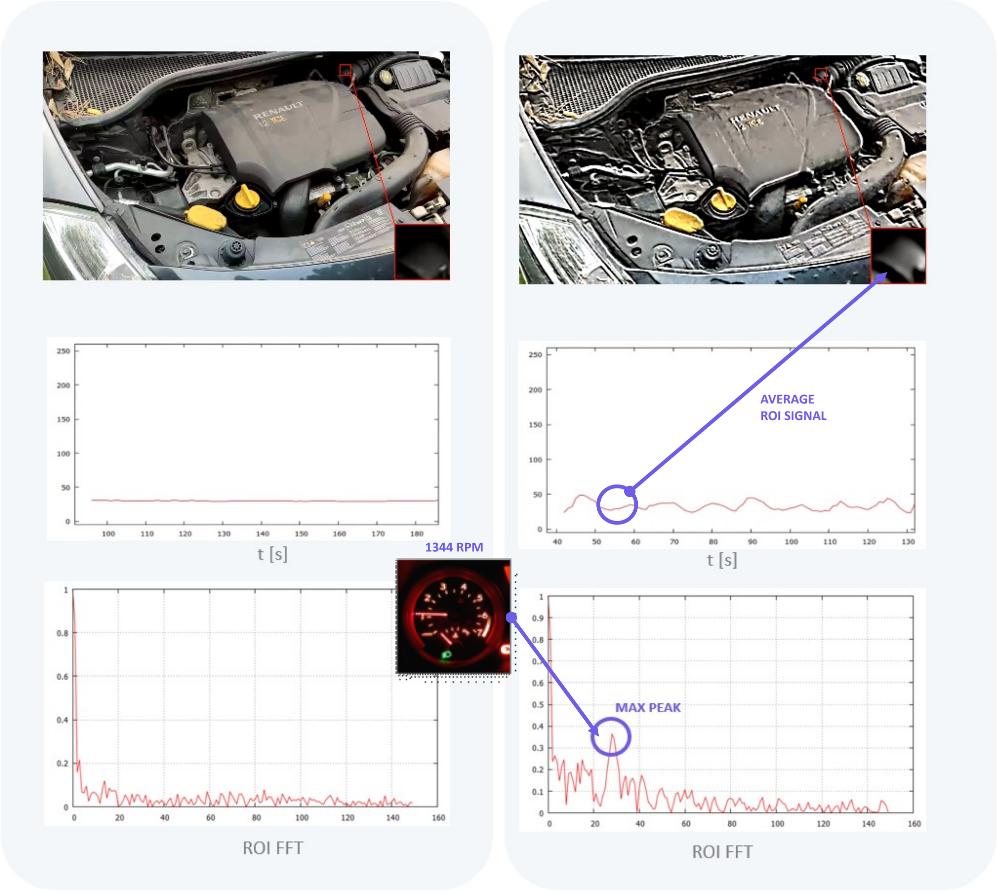


Figure 57: EVM applied to an engine. Results.

4.1.3 *Software and GUI*

The software is designed to have an easy frontend GUI (Figure 58) to help user better interact with the system The backend includes all the EVM algorithm, camera acquisition and frame management. In the GUI the user can start the analys and when finished, an indicator shows the detected frequency on the screen, which in the case of heart rate, for example, corresponds to BPM. The main functions of the software take advantage

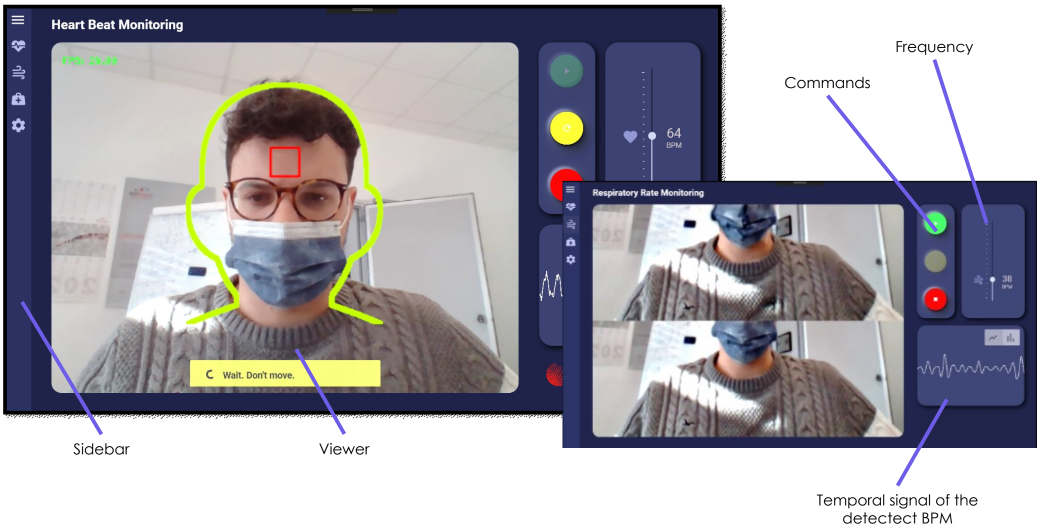


Figure 58: EVM GUI. It includes a viewer for the analyzed video that can be a live viewer if a camera (such as PC camera) is connected, commands buttons to start analysis, and plots for temporal and frequency signal detected.

of the *OpenCV* library. *OpenCV* is an open source computer vision library built to provide a common infrastructure for computer vision applications. The uses of OpenCV have different applications, and industrial use is also widespread. It is natively written in C++ but can be also used in Python, Java and MATLAB interfaces. Moreover it supports Windows, Linux, Android and Mac OS. *OpenCV* includes different functions for image processing, video processing, but also object detection, face detection, object tracking and machine learning.

The software consists of two macro modules:

- **Engine**: written in C++ language. It consists of the DLLs in with the all EVM algorithm;
- **GUI**: written in C# language. constitutes all the user interface components. The commands of the GUI will call the Engine functions in C++ that allow to connect with camera, start the EVM algorithm, send and receive data (such as video frames).

4.1.3.1 *Engine*

The engine is fully written in C++ to boost-up the performance. It includes 3 phases to manage the whole pipeline that start from a frame and produce

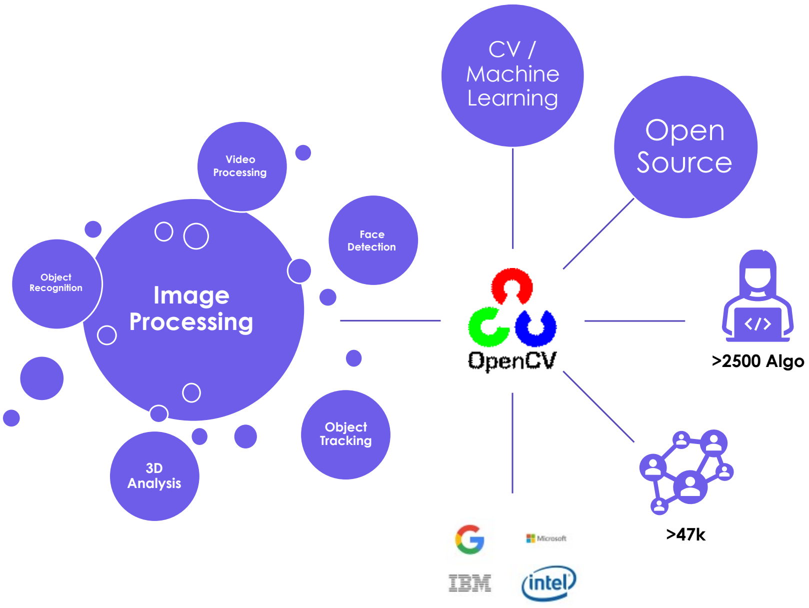


Figure 59: The OpenCV library. A collection of OpenCV's features set, those related to image processing and video processing were especially used.

the magnified video: pre-processing, processing, post-processing.

In the pre-processing and processing phases (Figure 60), the following operations are performed:

- an *OpenCV*-based function captures an image by a camera, in the case of heartbeat detection the internal camera of the PC is used;
- A ROI is selected through coordinates (size of the ROI x, y, w, h) passed from the GUI and is then converted from $uchar8[bit]$ format to $float32[bit]$, to avoid approximation errors;
- calculi to improve robustness are performed on the image, such as G/R ratio scaling factor (Equation 73): since the same light source affects all channels, this ratio produces a signal which is unaffected by external light;
- an image pyramind is calculated. The image is down-scaled, amplified by a temporal filter with the other frames, and then up-scaled. The result image is now added to the original one.

In the post-processing phase (Figure 61), the following operations are performed:

- a C++ class called "DataManipulator" acts as the manager for the ROI. It takes the analyzed ROI, averages the pixels over three channels, and obtains a 1-dimensional signal from it, which is then interpolated to reconstruct a cleaner signal;

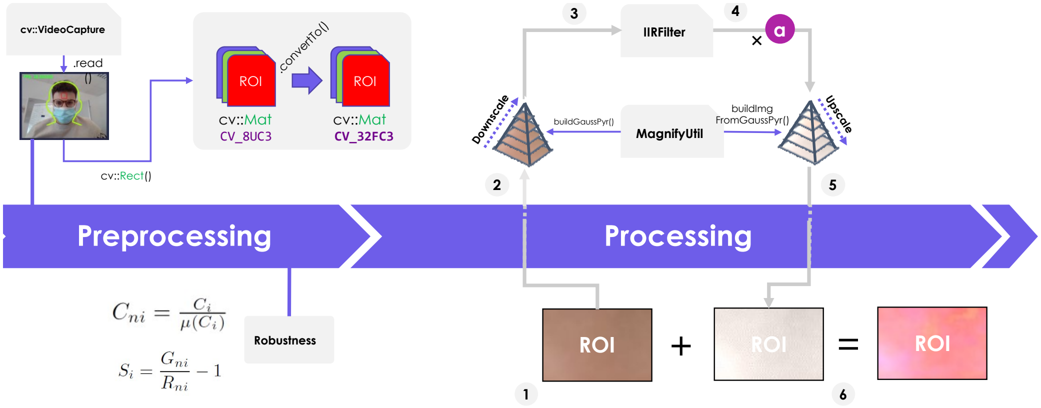
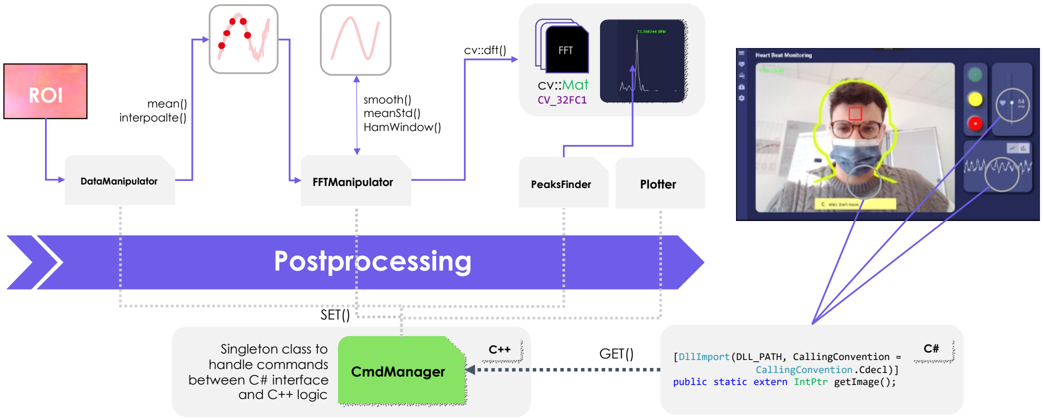


Figure 60: Pre-processing and processing pipeline. Read a ROI from the recorded video, make some robustness enhancement processing and apply the EVM on the ROI.

- another C++ class takes care of processing this signal and generating an FFT. A peaks finding function takes care of finding the maximum peak;
- these steps generate outputs to be passed to the GUI (result ROI, FFT, maximum peaks). A class "CmdManager" is called on the GUI side (in C#) in a C++-independent way.



```
[DllImport(DLL_PATH, CallingConvention =
                CallingConvention.Cdecl)]
public static extern IntPtr getImage();
```

Figure 61: Post-processing pipeline. Read the the processed ROI, extract a clean signal and produce a FFT with the main peaks which correspond to the detected frequencies in the video.

Since C# calls C++ DLL's functions when needed, the graphics side is decoupled from the engine side, which will not be slowed down. For example, the acquisition of each frame can occur internally in the PC camera at 30 FPS, and the EVM algorithm will process them at this rate. The user-side live video does not have to be at 30 FPS; in fact, it is up to the GUI side to decide at what frequency to go "fishing" for frames. This is done because the live video in the GUI only needs to give qualitative information, there is no need for all frames to be displayed; this also brings the advantage of keeping engine (C++) and GUI (C#) decoupled.

CONCLUSIONS

This thesis presents a comprehensive study on various strategies to enhance various aspects of manufacturing processes through the integration of two key technologies: robot control algorithms and computer vision. In the first part of the research, a series of control algorithms have been developed to expand and improve the field of Physical Human-Robot Interaction (pHRI). In the second part, quality control and analysis systems based on computer vision have been developed for use in the manufacturing industry and beyond.

Chapter 1 provides an introduction to the main technologies used and an overview of the algorithms or applications that have been developed throughout the thesis.

Chapter 2 delves into the description of two control strategies for pHRI settings. The first strategy is based on an elasto-plastic control in which a person exerts a force on a robot, causing its movement to deviate from a preset trajectory according to an elastic or plastic behaviour, depending on certain conditions. This is particularly useful in situations where it is necessary to perform different tasks quickly and in the same work area. The second control strategy exploits a totally offline human intention model to create a fuzzy-based control that scales the parameters of the robot's impedance based on the force and speed of the user's movement.

Chapter 3 illustrates a system based on a 3D scanner for performing defect checks on die castings. The procedure for carrying out the acquisition, alignment, calibration, and analysis of a point cloud produced by the scanner is detailed.

Chapter 4 illustrates an application created with the Eulerian Video Magnification (EVM) technique, which allows for the amplification of small vibrations present in a video signal, otherwise invisible to the human eye. This system can be applied in both the healthcare sector for the detection of heartbeat and frequency, as well as in industrial and manufacturing applications, such as estimating the vibration frequencies of a machine by means of a standard video of the machine itself.

Overall, this thesis demonstrates the potential of integrating robot control algorithms and computer vision in manufacturing processes to improve efficiency, accuracy, and productivity. The research work presented here lays the foundation for further advancements in the field of automation engineering and offers practical solutions for industry professionals to implement in their manufacturing processes.

BIBLIOGRAPHY

[1] Universal-Robots Official Page. URL https://www.universal-robots.com/.

[2] Robotiq Official Page. URL https://robotiq.com/.

[3] Reymus Marcel, Hickel Reinhard, and Keßler Andreas. Accuracy of cad/cam-fabricated bite splints: milling vs 3d printing. *Clinical oral investigations*, 24(12):4607–4615, 2020.

[4] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM transactions on graphics (TOG)*, 31(4):1–8, 2012.

[5] Mohd Aiman Kamarul Bahrin, Mohd Fauzi Othman, Nor Hayati Nor Azli, and Muhamad Farihin Talib. Industry 4.0: A review on industrial automation and robotic. *Jurnal teknologi*, 78(6-13), 2016.

[6] Michael Rüßmann, Markus Lorenz, Philipp Gerbert, Manuela Waldner, Jan Justus, Pascal Engel, and Michael Harnisch. Industry 4.0: The future of productivity and growth in manufacturing industries. *Boston consulting group*, 9(1):54–89, 2015.

[7] Günther Schuh, Till Potente, Cathrin Wesch-Potente, Anja Ruth Weber, and Jan-Philipp Prote. Collaboration mechanisms to increase productivity in the context of industrie 4.0. *Procedia Cirp*, 19:51–56, 2014.

[8] Elvis Hozdić. Smart factory for industry 4.0: A review. *International Journal of Modern Manufacturing Technologies*, 7(1):28–35, 2015.

[9] Dmitry Ivanov, Boris Sokolov, and Marina Ivanova. Schedule coordination in cyber-physical supply networks industry 4.0. *IFAC-PapersOnLine*, 49(12):839–844, 2016.

[10] Selim Erol, Andreas Jäger, PhiliFFpp Hold, Karl Ott, and Wilfried Sihn. Tangible industry 4.0: a scenario-based approach to learning for the future of production. *Procedia CiRp*, 54:13–18, 2016.

[11] Francisco Almada-Lobo. The industry 4.0 revolution and the future of manufacturing execution systems. *Journal of innovation management*, 3(4):16–21, 2015.

[12] Michael A. Goodrich and Alan C. Schultz. Human-robot interaction: A survey, 2007. ISSN 15513955.

[13] Antonio Bicchi, Michael A. Peshkin, and J. Edward Colgate. Safety for Physical Human–Robot Interaction. In *Springer Handb. Robot.*, pages 1335–1348. Springer Berlin Heidelberg, 2008. doi: 10.1007/978-3-540-30301-5_58.

[14] Federico Vicentini. Terminology in safety of collaborative robotics. *Robotics and Computer-Integrated Manufacturing*, 63:101921, 2020. ISSN 0736-5845. doi: https://doi.org/10.1016/j.rcim.2019.101921.

[15] Andrea Cherubini, Robin Passama, André Crosnier, Antoine Lasnier, and Philippe Fraisse. Collaborative manufacturing with physical human-robot interaction. *Robot. Comput. Integr. Manuf.*, 40:1–13, aug 2016. ISSN 07365845. doi: 10.1016/j.rcim.2015.12.007.

[16] Clint Heyer. Human-robot interaction and future industrial robotics applications. In *IEEE/RSJ 2010 Int. Conf. Intell. Robot. Syst. IROS 2010 - Conf. Proc.*, pages 4749–4754, 2010. ISBN 9781424466757. doi: 10.1109/IROS.2010.5651294.

[17] Fabrizio Caccavale, Christian Ott, Bernd Winkler, and Zachary Taylor. *Bringing Innovative Robotic Technologies from Research Labs to Industrial End-users*. Springer, 2019. ISBN 978-3-030-34506-8. doi: 10.1007/978-3-030-34507-5.

[18] Robin Roberson Murphy. Human-robot interaction in rescue robotics. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, 34(2):138–153, may 2004. ISSN 10946977. doi: 10.1109/TSMCC.2004.826267.

[19] Haoyong Yu, Sunan Huang, Gong Chen, Yongping Pan, and Zhao Guo. Human-Robot Interaction Control of Rehabilitation Robots with Series Elastic Actuators. *IEEE Trans. Robot.*, 31(5):1089–1100, oct 2015. ISSN 15523098. doi: 10.1109/TRO.2015.2457314.

[20] G. Legnani and I. Fassi. *Robotica industriale*. CEA Casa Editrice Ambrosiana, Milano, 2016.

[21] Neville Hogan. Impedance Control: An Approach to Manipulation. In *1984 Am. Control Conf.*, pages 304–313. IEEE, jul 1984. doi: 10.23919/ACC.1984.4788393.

[22] Prashant K. Jamwal, Shahid Hussain, Mergen H. Ghayesh, and Svetlana V. Rogozina. Impedance Control of an Intrinsically Compliant Parallel Ankle Rehabilitation Robot. *IEEE Trans. Ind. Electron.*, 63(6):3638–3647, jun 2016. ISSN 02780046. doi: 10.1109/TIE.2016.2521600.

[23] Zhijun Li, Zhicong Huang, Wei He, and Chun Yi Su. Adaptive impedance control for an upper limb robotic exoskeleton using biological signals. *IEEE Trans. Ind. Electron.*, 64(2):1664–1674, feb 2017. ISSN 02780046. doi: 10.1109/TIE.2016.2538741.

[24] Agostino De Santis, Bruno Siciliano, Alessandro De Luca, and Antonio Bicchi. An atlas of physical human–robot interaction. *Mech. Mach. Theory*, 43(3):253–270, 2008.

[25] Gennaro Raiola, Carlos Alberto Cardenas, Tadele Shiferaw Tadele, Theo De Vries, and Stefano Stramigioli. Development of a safety-and energy-aware impedance controller for collaborative robots. *IEEE Robotics and automation letters*, 3(2):1237–1244, 2018.

[26] Fanny Ficuciello, Luigi Villani, and Bruno Siciliano. Impedance control of redundant manipulators for safe human-robot collaboration. *Acta Polytechnica Hungarica*, 13(1):223–238, 2016.

[27] Bakur Alqaudi, Hamidreza Modares, Isura Ranatunga, Shaikh M Tousif, Frank L Lewis, and Dan O Popa. Model reference adaptive impedance control for physical human-robot interaction. *Control Theory and Technology*, 14(1):68–82, 2016.

[28] Curt A Laubscher and Jerzy T Sawicki. A robust impedance controller for improved safety in human–robot interaction. *Journal of Dynamic Systems, Measurement, and Control*, 143(9), 2021.

[29] Peng Song, Yueqing Yu, and Xuping Zhang. A tutorial survey and comparison of impedance control on robotic manipulation. *Robotica*, 37(5):801–836, 2019.

[30] Fanny Ficuciello, Luigi Villani, and Bruno Siciliano. Variable Impedance Control of Redundant Manipulators for Intuitive Human-Robot Physical Interaction. *IEEE Trans. Robot.*, 31(4):850–863, 2015. ISSN 15523098. doi: 10.1109/TRO.2015.2430053.

[31] Loris Roveda, Nicola Pedrocchi, and Lorenzo Molinari Tosatti. An Adaptive Manual Guidance Formulation with Force-Tracking Control for Industrial Cooperative Tasks. (August), 2016.

[32] Jan F Broenink and Martin L J Tiernego. Peg-in-Hole assembly using Impedance Control with a 6 DOF Robot. *Proc. 8th Eur. Simul. Symp.*, (January):504–508, 1996.

[33] Jonas Buchli, Evangelos Theodorou, Freek Stulp, and Stefan Schaal. Variable impedance control: A reinforcement learning approach. *Robot. Sci. Syst.*, 6(1):153–160, 2011. ISSN 2330765X.

[34] T. Tsumugiwa, R. Yokogawa, and K. Hara. Variable impedance control with regard to working process for man-machine cooperation-work system. In *Proc. 2001 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Expand. Soc. Role Robot. Next Millenn. (Cat. No.01CH37180)*, volume 3, pages 1564–1569. IEEE. ISBN 0-7803-6612-3. doi: 10.1109/IROS.2001.977202.

[35] Dachao Mao, Wenlong Yang, and Zhijiang Du. Fuzzy variable impedance control based on stiffness identification for human-robot cooperation. In *IOP Conf. Ser. Earth Environ. Sci.*, volume 69. Institute of Physics Publishing, jun 2017. doi: 10.1088/1755-1315/69/1/012090.

[36] Toru Tsumugiwa, Ryuichi Yokogawa, and Kei Hara. Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 1, pages 644–650. IEEE, 2002.

[37] MM Rahman, R Ikeura, and K Mizutani. Investigating the impedance characteristic of human arm for development of robots to co-operate with human operators. In *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, volume 2, pages 676–681. IEEE, 1999.

[38] Vincent Duchaine and Clement M Gosselin. General model of human-robot cooperation using a novel velocity based variable impedance control. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, pages 446–451. IEEE, 2007.

[39] Vincent Duchaine and Clement M Gosselin. Investigation of human-robot interaction stability using lyapunov theory. In *2008 IEEE International Conference on Robotics and Automation*, pages 2189–2194. IEEE, 2008.

[40] Fotios Dimeas and Nikos Aspragathos. Fuzzy Learning Variable Admittance Control for Human-Robot Cooperation. Technical report.

[41] Kazuo Kiguchi and Yoshiaki Hayashi. An emg-based control for an upper-limb power-assist exoskeleton robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(4):1064–1071, 2012.

[42] Baoping Yuan Masashi Sekine,. Variable Impedance Control Based on Impedance Estimation Model with EMG Signals during Extension and Flexion Tasks for a Lower Limb Rehabilitation Robotic System. *J. Nov. Physiother.*, 03(05), 2013. doi: 10.4172/2165-7025.1000178.

[43] Duane McRuer. Human dynamics in man-machine systems. *Automatica*, 16(3):237–253, 1980.

[44] Niko Herakovic. *Robot vision in industrial assembly and quality control processes*. INTECH Open Access Publisher London, UK, 2010.

[45] Ladislav Morovič, Juraj Vagovskỳ, and Ivan Buranskỳ. Shape investigation of worn cutting inserts with utilization of active triangulation. In *Key Engineering Materials*, volume 581, pages 22–25. Trans Tech Publ, 2014.

[46] Vopát Tomas, Peterka Jozef, Kováč Mario, and Buranskỳ Ivan. The wear measurement process of ball nose end mill in the copy milling operations. *Procedia Engineering*, 69:1038–1047, 2014.

[47] Ismael Colomina and Pere Molina. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS Journal of photogrammetry and remote sensing*, 92:79–97, 2014.

[48] Andreas Kolb, Erhardt Barth, Reinhard Koch, and Rasmus Larsen. Time-of-flight cameras in computer graphics. In *Computer Graphics Forum*, volume 29, pages 141–159. Wiley Online Library, 2010.

[49] João Guilherme DM França, Mário A Gazziro, Alessandro Noriaki Ide, and José Hiroki Saito. A 3d scanning system based on laser triangulation and variable field of view. In *IEEE International Conference on Image Processing 2005*, volume 1, pages I–425. IEEE, 2005.

[50] M Martorella, F Salvetti, and D Stagliano. 3d target reconstruction by means of 2d-isar imaging and interferometry. In *2013 IEEE Radar Conference (RadarCon13)*, pages 1–6. IEEE, 2013.

[51] revopoint3d Official Page. URL `https://www.revopoint3d.com/how-does-light-influence-3d-scanning/`.

[52] Juraj Vagovskỳ, Ivan Buranskỳ, and Augustín Görög. Evaluation of measuring capability of the optical 3d scanner. *Procedia Engineering*, 100:1198–1206, 2015.

[53] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics express*, 18(10):10762–10774, 2010.

[54] Wim Verkruysse, Lars O Svaasand, and J Stuart Nelson. Remote plethysmographic imaging using ambient light. *Optics express*, 16(26): 21434–21445, 2008.

[55] Ce Liu, Antonio Torralba, William T Freeman, Frédo Durand, and Edward H Adelson. Motion magnification. *ACM transactions on graphics (TOG)*, 24(3):519–526, 2005.

[56] Koichi Hashiguchi. *Elastoplasticity theory*, volume 42. Springer, 2009.

[57] Ettore Pennestrì, Valerio Rossi, Pietro Salvini, and Pier Paolo Valentini. Review and comparison of dry friction force models. *Nonlinear dynamics*, 83(4):1785–1801, 2016.

[58] Phil R Dahl. A solid friction model. Technical report, Aerospace Corp El Segundo Ca, 1968.

[59] C. Canudas de Wit, P. Lischinsky, K. J. Åström, and H. Olsson. A New Model for Control of Systems with Friction. *IEEE Trans. Automat. Contr.*, 40(3):419–425, 1995. ISSN 15582523. doi: 10.1109/9.376053.

[60] Karl Johan Åström and Carlos Canudas de Wit. Revisiting the LuGre friction model To cite this version : HAL Id : hal-00394988 Revisiting the LuGre Model Stick-slip motion and rate dependence. 28(6):101–114, 2009.

[61] E Velenis, P Tsiotras, and C. Canudas-de Wit. Extension of the lugre dynamic tire friction model to 2d motion. *Proc. 10th IEEE Mediterr. Conf. Control Autom.*, (January 2002):9–12.

[62] Pierre Dupont, Brian Armstrong, and Vincent Hayward. Elasto-plastic friction model: Contact compliance and stiction. *Proc. Am. Control Conf.*, 2:1072–1077, 2000. ISSN 07431619. doi: 10.1109/acc.2000.876665.

[63] Anis Koubâa et al. *Robot Operating System (ROS).*, volume 1. Springer, 2017.

[64] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[65] Andrea Bajo and Nabil Simaan. Hybrid motion/force control of multi-backbone continuum robots. *Int. J. Rob. Res.*, 35(4):422–434, apr 2016. ISSN 17413176. doi: 10.1177/0278364915584806.

[66] Aghil Jafari and Jee Hwan Ryu. Independent force and position control for cooperating manipulators handling an unknown object and interacting with an unknown environment. *J. Franklin Inst.*, 353 (4):857–875, mar 2016. ISSN 00160032. doi: 10.1016/j.jfranklin.2015.12.010.

[67] Christian Ott, Ranjan Mukherjee, and Yoshihiko Nakamura. Unified impedance and admittance control. In *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 554–561, 2010. ISBN 9781424450381. doi: 10.1109/ROBOT.2010.5509861.

[68] Ryojun Ikeura and Hikaru Inooka. Variable impedance control of a robot for cooperation with a human. In *Proc. - IEEE Int. Conf. Robot. Autom.*, volume 3, pages 3097–3102, 1995. ISBN 0780319656. doi: 10.1109/ROBOT.1995.525725.

[69] Joaquin A. Blaya and Hugh Herr. Adaptive Control of a Variable-Impedance Ankle-Foot Orthosis to Assist Drop-Foot Gait. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 12(1):24–31, mar 2004. ISSN 15344320. doi: 10.1109/TNSRE.2003.823266.

[70] Abhishek Gupta and Marcia K. Robotic Exoskeletons for Upper Extremity Rehabilitation. In *Rehabil. Robot.* I-Tech Education and Publishing, aug 2007. doi: 10.5772/5171.

[71] Robert Riener, Martin Frey, Michael Bernhardt, Tobias Nef, and Gery Colombo. Human-centered rehabilitation robotics. In *Proc. 2005 IEEE 9th Int. Conf. Rehabil. Robot.*, volume 2005, pages 319–322, 2005. ISBN 0780390032. doi: 10.1109/ICORR.2005.1501110.

[72] Bruno Siciliano and Luigi Villani. *Robot Force Control*. Springer US, Boston, MA, 1999. ISBN 978-1-4613-6995-0. doi: 10.1007/978-1-4615-4431-9.

[73] Tijana T Ivancevic, Bojan N Jovanovic, Sasha A Jovanovic, Leon Lukman, Alexandar Lukman, and Milka Djukic. Linear versus nonlinear human operator modeling. *International Journal of Mathematics, Game Theory, and Algebra*, 20(2):123, 2011.

[74] Duane T McRuer and Ezra S Krendel. The human operator as a servo system element. *Journal of the Franklin Institute*, 267(6):511–536, 1959.

[75] Duane T McRuer, David H Weir, Henry R Jex, Raymond E Magdaleno, and R Wade Allen. Measurement of driver-vehicle multiloop response properties with a single disturbance input. *IEEE Transactions on Systems, Man, and Cybernetics*, (5):490–497, 1975.

[76] Miguel Martínez-García, Timothy Gordon, and Lei Shu. Extended crossover model for human-control of fractional order plants. *IEEE Access*, 5:27622–27635, 2017.

[77] Duane T McRuer and Henry R Jex. A review of quasi-linear pilot models. *IEEE transactions on human factors in electronics*, (3):231–249, 1967.

[78] Duane T. McRuer. Mathematical Models of Human Pilot Behavior. Technical report, 1974.

[79] Photoneo Official Page. URL https://www.photoneo.com/phoxi-3d-scanner.

[80] Paolo Minetola, Flaviana Calignano, and Luca Iuliano. *Analysis of alignment procedures for contactless inspection of rapid manufactured parts*. na, 2010.

[81] Paolo Minetola. The importance of a correct alignment in contactless inspection of additive manufactured parts. *International journal of precision engineering and manufacturing*, 13(2):211–218, 2012.

[82] Jue Wang, Steven M Drucker, Maneesh Agrawala, and Michael F Cohen. The cartoon animation filter. *ACM Transactions on Graphics (TOG)*, 25(3):1169–1173, 2006.

[83] Edward H Adelson, Charles H Anderson, James R Bergen, Peter J Burt, and Joan M Ogden. Pyramid methods in image processing. *RCA engineer*, 29(6):33–41, 1984.

[84] Luowen Li, Lihua Xie, Wei-Yong Yan, and Yeng Chai Soh. Design of low-order linear-phase iir filters via orthogonal projection. *IEEE transactions on signal processing*, 47(2):448–457, 1999.

[85] Buon Kiong Lau, Victor Sreeram, Yee Hong Leung, and Kok Lay Teo. Design of low order approximately linear phase iir filters. In *1998 IEEE Symposium on Advances in Digital Filtering and Signal Processing. Symposium Proceedings (Cat. No. 98EX185)*, pages 92–95. IEEE, 1998.